

*Ассоциация участников технологических кружков*

УТВЕРЖДАЮ  
Президент Ассоциации



А.И. Федосеев

Дополнительная общеразвивающая образовательная программа

**«Технологии дополненной реальности»**

Возраст учащихся: 11-17 лет

Срок реализации программы: 1 год

Разработчик: А.Г. Балахчи, З.А. Воловикова,  
Ю.В. Пестова, Я.А. Угорская, М.А. Сокольская  
команда профиля  
«Технологии дополненной реальности» НТО  
Ассоциация участников технологических кружков

Иркутск, 2021 г.

## Содержание

Требования к формированию команд	7
<b>Планируемые результаты программы</b>	<b>11</b>
<b>Учебно-тематическое планирование</b>	<b>13</b>
Содержание программы	14
Контрольно-измерительные материалы курса	27
Материально-техническое оснащение кружка	35
Методические рекомендации к проведению первого раздела программы	36
Методические рекомендации по проведению второго раздела программы.	37
Методические материалы курса	37
Первый модуль индивидуального этапа	37
Задача 1. Что такое куча?	38
Задача 2. Нужны ли городу развлекательные центры?	39
Задача 3. Как найти проходное место в городе?	44
Задача 4. Маршруты	46
Задача 5. Анализ сообщений в социальных сетях	48
Задача 6. Погода в городе	51
Задача 7. Выбираем место под новый жилой комплекс	56
Задача 8. Время работы развлекательного центра	57
Задача 9. Сколько "счастья" надо?	60
Задача 10. Конкурс заявок	65
Второй модуль индивидуального этапа.	67
Постановка проблемы для участников.	67
Задача 2. Единицы измерения площадей земельных участков	73
Задача 3. Парковые зоны	73
Задача 4. Замечательные места	76
Задача 5. "Фишки" в облаках	78
Задача 6. Предпочтения	86
Задача 7. Удобство в дорогах	88
Задача 8. Центр.	91
Задача 9. Центральная достопримечательность	93
Задача 10. Время работы	94
Третий модуль индивидуального этапа.	96
Постановка проблемы для участников.	96
Задача 1. Сбор данных для исследований	97
Задача 2. Анализ отзывов	100
Задача 3. Анализ целевой аудитории района	102
Задача 4. Какие в районе проблемы?	104
Задача 5. Формирование рейтинга мест	107

Задача 6. Что важнее?	109
Задача 7. Умные решения	110
Задача 8. Плотность предприятий в районе	114
Задача 9. Где открыть ресторан?	117
Задача 10. Ресторан или кафе быстрого питания?	124
Задачи командного этапа	128
Легенда.	128
Генеральная задача командного этапа.	128

## Пояснительная записка

На состояние здоровья человека влияет образ жизни и окружающая среда, поэтому так важно, где и как он живет. Объекты городской инфраструктуры обладают визуальными и смысловыми кодами, которые оказывают влияние на наше эмоциональное состояние, а в целом определяют поведение социума на данной территории. Каждая улица города имеет свое настроение, которое вносит свой вклад в культурный и социальный фон всей территории. Чтобы создать комфортную для человека среду обитания, важно исследовать ее состояние, правильно выделять негативные и позитивные факторы, принимать управляющие решения, которые основываются на сборе информации и анализе разнообразных данных об объектах уличного ландшафта. Современные исследования среды обитания очень часто выполняются на стыке компьютерных и гуманитарных наук, касаются понятий «цифровая гуманитаристика», «социальная инженерия», урбанистика и дизайн окружающей среды.

Материалы курса, предлагаемые командой разработчиков, фокусируют внимание школьников, наставников, родителей и преподавателей на исследованиях, задействующих самые разные науки. При этом планируется задействовать уже привычные всем технологии программирования для сбора, обработки, анализа и визуализации пакетов данных. Показать их использование

в новом контексте, заинтересовать подходами к решению проблем, регулированию эмоционального фона общества, повышения комфортности окружающей среды для населения некоторой территории.

В процессе решения комплекса задач, учащиеся работают с таким компонентом как “вычислительное эссе”. Применение такого образовательного компонента позволит участникам по-новому взглянуть на формирование вычислительной культуры обучающихся, будет способствовать формированию умений выражать свои мысли, строить гипотезы, описывать подходы, презентовать материалы. Вычислительное эссе — это симбиоз сочинения и сложной алгебраической задачи, вычислительная история с инфографикой: сюжет надо придумать, главных героев просчитать, а кульминацию и развязку визуализировать. Термин “вычислительное эссе” впервые ввел Стивен Вольфрам.

Вычислительное эссе – итоговый продукт командных соревнований, форма публикации, объединяющая выводы и позиции авторов, вычисления (снабженные комментариями), интерактивную инфографику. Вычислительное эссе, позволяет добавлять и исправлять данные в публикации по мере поступления или корректировки.

В качестве ключевой технологий реализации вычислительного эссе разработчики предложили использовать Google Collab.

Google Colab — это облачный сервис, направленный на упрощение исследований в области машинного обучения и проведения компьютерных вычислений. Можно сказать, что Colab является некоторым аналогом гугл-документов для JupyterNotebook.

Ученикам 5–7 классов школ, командам участникам курса «Технологии для среды обитания» предстоит провести и оформить интерактивную публикацию исследования по теме «Настроение улиц города».

Ребята, в процессе подготовки эссе решают следующие обобщенные задачи:

1) сбор из свободных источников информации данных об объектах инфраструктуры улиц города (или любой другой территории проживания), включающие:

- цветовые решения;
- естественное и искусственное освещение;
- строительные материалы;
- архитектурные формы;
- функциональное назначение зданий;
- озеленение и т. п.

2) формулировка своей гипотезы об удобстве, полезности, востребованности изученных объектов и аргументация выводов;

3) обработка данных, построение графиков, подтверждающих или опровергающих выдвинутые ими ранее гипотезы. Все работы этой части выполняются с использованием облачного сервиса Google Collab;

4) оформление материалов в виде вычислительного эссе с интерактивной инфографикой.

Содержание опубликованных материалов эссе меняется динамически. Таким образом, с одной стороны, мы получаем инструмент для исследований, а с другой, учебное пособие, используя которое, участники на живых примерах изучают инструменты вычислительной математики и анализа данных.

Цель программы: формирование навыков учащихся по решению задач, направленных на поиск способов решения проблем, связанных с организацией комфортной среды для жизни и деятельности сообществ людей.

Задачи программы:

- Знакомство учащихся с алгоритмами поиска оптимальных решений в вопросах организации среды обитания
- Формирование навыков учащихся по анализу разрозненных данных, выявлением неочевидных закономерностей.

**Возраст детей:** 5-7 класс

**Количество часов:** 2 академических часа в неделю (1 занятие по 2 часа)

**Сроки реализации программы:** 72 часа

### **Особенности реализации курса**

Программа обучения содержит два основных модуля. При этом задачи для каждого раздела подобны по содержанию и структуре, однако второй раздел предполагает большую степень самостоятельности школьников и сильнее ограничен во времени.

Первый раздел программы “Технологии для среды обитания” содержит не только задачи, их решения и наборы входных данных, но и значительное количество обучающих материалов. Обучающие материалы необходимы, т. к. учащиеся 5–7 классов сталкиваются с ситуациями, которые не входят в школьную программу, соответственно, у учащихся отсутствуют знания, умения, позволяющие решать задачи данного раздела.

Слушатели курса получают задачи и тестовые задания по следующим модулям:

- программирование на языке Python;
- сбор, обработка, анализ данных (все с учетом школьной образовательной программы для соответствующего возраста);
- визуализация и инфографика;
- сторителлинг;

- системы коллективного пользования (куда войдут, в том числе, задания по системе Google Colab).

По результатам первого учебного раздела составляются рейтинги участников, в соответствии с которым необходимо будет сформировать команды для второго раздела обучения. Поскольку каждая задача первого раздела рассчитана на определенную роль участника в команде, то рейтинг составляется по каждому модулю отдельно (что, возможно, поможет после первого раздела распределить ребят по ролям). Все школьники начинают в равных условиях и роль участника в его будущей команде определяет педагог по результатам анализа всех рейтинговых таблиц. Здесь важно заметить, что первый раздела обучения индивидуален, команды формируются только для второго раздела.

Второй раздел обучения включает в себя три основных этапа, на каждом из которых учащиеся получают новую информацию, тренируют новые навыки. В итоге команды, на заключительном занятии представляют жюри (в состав которого могут входить как педагоги, так и учащиеся старших классов, а также представители местного сообщества, эксперты в своих вопросах) свои вычислительные эссе.

Лучшие вычислительные эссе команд могут быть рекомендованы к публикациям и представлению на научно-практических конференциях школьников, таких как: «Ломоносов», «Шаг в будущее», МНСК, Удивительный мир и другие. На основе изложенных материалов может быть собран элективный курс по началам анализа данных для учащихся средних классов школ.

## **Требования к формированию команд**

Каждая команда учащихся состоит из 4 человек: программист, аналитик, дизайнер-инфографик, журналист данных.

**Программист** - знает основы программирования на языке Python.

**Аналитик** – владеет основными математическими методами обработки числовых данных, построения графиков, знаком с понятием функции, основами теории вероятностей и комбинаторики, базовыми методами оптимального выбора.

**Дизайнер-инфографик** – понимает значение и сущность понятия «график», владеет компьютерными технологиями построения графиков, любым компьютерным графическим редактором (GIMP, Inscapе, Photoshop, CorelDraw).

**Журналист данных** - обладает широким кругозором, способен к оригинальному мышлению, построению гипотез, владеет письменной речью - умеет излагать свои мысли, формировать логическую цепочку: утверждение, аргументы, вывод.

В таблице 1 можно посмотреть роли участников и выдвигаемые к ним требования.

*Таблица 1. Роли участников команд*

№	Роль в команде	Задача	Компетенции
1	Программист (пишет программный код: обрабатывает данные)	Данные собраны АНАЛИТИКОМ. Теперь ин надо обработать в соответствии с выдвинутой командой гипотезой. Участник пишет код для обработки данных: вычисления, поиск минимальных и максимальных значений, построение графиков, создание	<b>Софт:</b> логическое мышление, Коммуникабельность, умение работать в команде, пунктуальность, стрессоустойчивость <b>Хард:</b> знает основы программирования на языке Python



		интерактивных манипуляторов для обработки данных <b>АНАЛИТИКОМ</b>	<b>Возраст:</b> 6-7 класс <b>Дублер для роли:</b> аналитика
2	Аналитик (производит предварительные расчеты, готовит структуры данных для программиста)	Собирает первичные данные согласно условию задачи. Формирует логику решения. Строит предположения относительно выдвинутых гипотез. Может получить решение задачи аналитически, опираясь на математические методы. Оценивает правильность, полученного <b>ПРОГРАММИСТОМ</b> численного решения или графика.	<b>Софт:</b> логическое мышление, Коммуникабельность, умение работать в команде, креативность, наблюдательность <b>Хард:</b> владеет основными математическими методами обработки числовых данных, построения графиков, знаком с понятием функции, основами теории вероятностей и комбинаторики, базовыми методами оптимального выбора <b>Возраст:</b> 5-7 класс <b>Дублер для роли:</b> программист; журналист данных
3	Дизайнер инфографики (обрабатывает графики, подготовленные программистом создает визуальные истории из данных, оформляет публикацию)	На финале редактирует карты, схемы, графики, готовит иллюстрации для публикации. Отвечает за визуальное оформление публикации. Оформляет в графическом редакторе, полученные <b>ПРОГРАММИСТОМ</b> графики добавляя к инфографике подписи и графические элементы. Отвечает за стилевое оформление публикации.	<b>Софт:</b> креативность, художественный вкус, коммуникабельность, умение работать в команде <b>Хард:</b> понимает значение и сущность понятия «график», владеет компьютерными технологиями построения графиков, любым компьютерным графическим редактором (GIMP, Inscapе, Photoshop,

			CorelDraw) <b>Возраст:</b> 4-7 класс <b>Дублер для роли:</b> журналист данных
4	Журналист данных (формирует содержание публикации на основе анализа данных всей команды. Помогает сформулировать гипотезы и аргументы, сделать выводы)	Готовит сценарий и содержание публикации. Формулирует идею, логику, аргументы, выводы, обосновывает оригинальность и определяет перспективы. Важной частью является выполнение задач, связанных с анализом текстов. Журналист пишет текстовую часть эссе, соответственно владея навыками устной и письменной коммуникации лучше остальных членов команды. Поскольку основной задачей является интерактивная публикация исследования среды обитания, основанная на анализе данных, то <b>ЖУРАНЛИСТ</b> - режиссер всего процесса.	<b>Софт:</b> широкое мировоззрение, наблюдательность, креативность коммуникативность, умение работать в команде, эмоциональный интеллект <b>Хард:</b> системное мышление, способен к оригинальному мышлению, построению гипотез, владеет письменной речью - умеет излагать свои мысли, формировать логическую цепочку: утверждение, аргументы, вывод. Умеет выделять главное в тексте: видит проблему. Владеет методами построения облака слов. <b>Возраст:</b> 4-7 класс <b>Дублер для роли:</b> дизайнер инфографики.

Выполняя задания самостоятельно первого раздела, ребятам удастся получить необходимые для участия во втором разделе компетенции, определиться, какую роль в команде, он хочет выполнять. Это означает, что каждое задание будет снабжено обучающей статьей или ссылками на соответствующие учебные материалы.

## Планируемые результаты программы

### **Предметные результаты обучения:**

иметь представление об анализе данных и том, как аппарат анализа данных можно использовать в различных сферах;

иметь представление о том, как можно использовать данные из открытых источников для создания комфортной среды обитания человека; как учитывать данные в развитии человека и среды вокруг него;

иметь представление о том, что такое вычислительное эссе и интерактивные тетради Google Colab;

иметь представление о парсинге: извлечении данных из открытых источников, в том числе сайтов;

уметь вычислять и анализировать данные при помощи электронных таблиц Excel, строить графики;

уметь использовать инструменты облачного сервиса Google Colab;

уметь программировать на языке Python: вычислять, строить статические и динамические диаграммы, облака слов;

уметь на базовом уровне работать с геоданными: получать, обрабатывать их, наносить на карту;

иметь представления и уметь на базовом уровне анализировать изображения, например, при помощи специальных сервисов;

уметь работать с графическими редакторами: обрабатывать изображения (графики), рисовать иконки, оформлять карту.

### **Метапредметные результаты обучения:**

уметь использовать данные в исследовательской работе: определять тренды, строить гипотезы, подбирать аргументы на основе данных, подтверждающие или опровергающие гипотезы, формировать выводы;

развить познавательные интересы, интеллектуальные и творческие способности путем освоения и использования методов анализа данных;

приобрести опыт работы с данными в индивидуальной, групповой и коллективной учебно-познавательной и исследовательской деятельности, использовать этот опыт на практике в повседневной жизни.

### **Личностные результаты обучения:**

личностное и предпрофессиональное самоопределение через познавательную мотивацию к получению профессий, связанных с наукой о данных, программированием, дата-аналитикой, инфографикой и т.п.;

построение дальнейшей индивидуальной образовательной траектории через получение представления о перспективных развития науки о данных и инструментах, методиках и сферах практического применения data science;

осознание стратегической важности для государства, общества и для своего будущего развития науки о данных, понимание, что умение работать с данными является компетенцией, определяющей уровень грамотности каждого человека.

## Учебно-тематическое планирование

№	Тема	Количество часов	Лекции	Практика	Контроль
Раздел 1. Индивидуальное решение задач					
<b>1</b>	<b>Модуль 1.</b>	<b>20</b>	<b>8,5</b>	<b>11,5</b>	
1.1	Введение. Постановка проблемы для учащихся. Создание тетрадки. Задача. Что такое куча?	2	1	1	Задача
1.2	Построение столбчатых графиков Задача 2. Нужны ли городу развлекательные центры?	2	1	1	Задача
1.3	Вычисления в тетрадке на языке Python	2	1	1	
1.4	Задача 3. Как найти проходное место в городе? Задача 4. Маршруты	2	0,5	1,5	Задача
1.5	Построение облака слов Задача 5. Анализ сообщений в социальных сетях	2	1	1	Задача
1.6	Задача 6. Погода в городе	2	1	1	Задача
1.7	Построение линейчатых графиков Задача 7. Выбираем место под новый жилой комплекс	2	1	1	Задача
1.8	Задача 8. Время работы развлекательного центра	2	0,5	1,5	Задача
1.9	Построение круговых-диаграмм	2	1	1	
1.10	Задача 9. Сколько "счастья" надо? Задача 10. Конкурс заявок	2	0,5	1,5	Задача
<b>2</b>	<b>Модуль 2.</b>	<b>18</b>	<b>4,5</b>	<b>13,5</b>	
2.1	Введение. Постановка проблемы Задача 1. Ищем аналоги	2	0,5	1,5	Задача
2.2	Задача 2. Единицы измерения площадей земельных участков	2	0,5	1,5	Задача
2.3	Задача 3. Парковые зоны	2	0,5	1,5	Задача
2.4	Задача 4. Замечательные места	2	0,5	1,5	Задача
2.5	Задача 5. "Фишки" в облаках Задача 6. Предпочтения	2	0,5	1,5	Задача
2.6	Задача 7. Удобство в дорогах	2	0,5	1,5	Задача
2.7	Задача 8. Центр. Задача 9. Центральная достопримечательность	2	0,5	1,5	Задача
2.8	Задача 10. Время работы	4	1	3	Задача
<b>3</b>	<b>Модуль 3.</b>	<b>18</b>	<b>6</b>	<b>12</b>	
3.1	Введение. Постановки проблемы. Парсинг данных	2	1,5	0,5	Задача
3.2	Задача 1. Сбор данных для исследований	2	0,5	1,5	
3.3	Задача 2. Анализ отзывов	2	0,5	1,5	Задача
3.4	Задачи на работу с массивами и списками.	2	1	1	
3.5	Задача 3. Анализ целевой аудитории района	2	0,5	1,5	Задача
3.6	Задача 4. Какие в районе проблемы? Задача 5. Формирование рейтинга мест	2	0,5	1,5	Задача
3.7	Задача 6. Что важнее? Задача 7. Умные решения	2	0,5	1,5	Задача
3.8	Задача 8. Плотность предприятий в районе	2	0,5	1,5	Задача

3.9	Задача 9. Где открыть ресторан? Задача 10. Ресторан или кафе быстрого питания?	2	0,5	1,5	Задача
Раздел 2. Командное решение задач					
<b>4</b>	<b>Модуль 4.</b>	<b>16</b>	<b>3</b>	<b>13</b>	
4.1	Введение. Знакомство с Легендой. Решение организационных вопросов. Формирование команд	2	1	1	
4.2	Решение задач первого этапа	2		2	
4.3	Решение задач второго этапа	8	2	6	
4.4	Оформление работы	2		2	
4.5	Финал. Презентация вычислительных эссе	2		2	Проект
	<b>Итого:</b>	<b>72</b>	<b>22</b>	<b>50</b>	

## Содержание программы

Наименование, разделов, тем	Содержание	Виды учебных занятий, учебных работ
Раздел 1. Индивидуальное решение задач		
Модуль 1. Постановка проблемы для учащихся.	Мэр города Энска достроил новый развлекательный центр с парком аттракционов, и надеялся, что это место будет центром отдыха для горожан и гостей города. Необходимо провести исследования по улучшению качества инфраструктуры территории за счет ее трансформации, чтобы в городе появился еще один район с комфортной средой обитания.	Лекция
Задача. Что такое куча?	Создание тетрадки, настройки доступа к тетрадке. Разметка тетрадки текстовыми блоками. Правила безопасности при работе с тетрадкой	Лекция
	Написать программу, которая будет определять количество людей на квадратный метр — это покажет, сколько в среднем людей мы можем ожидать в центре отдыха с определенной площадью	Практика

Задача 2. Нужны ли городу развлекательные центры?	Построение столбчатых графиков	Лекция
	Дано количество людей - посетителей развлекательных центров в разных районах города в период с января по декабрь 2019 года. Необходимо- построить столбчатую диаграмму и ответить на вопрос, какой буквой закодирован торговый центр с наименьшей посещаемостью.	Практика
Задача 3. Как найти проходное место в городе?	Вычисления в тетрадке на языке Python	Лекция
	Проанализировать трехмерный массив, в котором по оси $Z$ - время в городе; $X$ , $Y$ - позиция в городе. Необходимо усреднить данные по времени. Ваша задача найти ячейку в полученном массиве с максимальным значением — это точка города, где за сутки проходит больше всего людей - "проходное" место.	Практика
Задача 4. Маршруты	Задана схема маршрутов и длина путей между пунктами. Известно, что точка $E$ — это область, где живет большинство людей. $Ж$ - исследуемый торговый центр. Определите количество маршрутов из района с максимальным населением до развлекательного центра, укажите длину наименьшего из найденных маршрутов.	Практика
Задача 5. Анализ сообщений в социальных сетях	Построение облака слов	Лекция
	Необходимо посмотреть сообщения в социальных сетях, оставленные людьми с улиц, расположенных по соседству с новым развлекательным центром улиц.	Практика

	Постройте облако слов на основе обобщенного текста, составленного из таких сообщений.	
Задача 6. Погода в городе	Постройте диаграмму-пирог, на которой можно посмотреть, соотношение ненастных дней и дней с хорошей погодой. Используйте два массива: в первом массиве $t$ - температура воздуха за день, во втором $w$ - количество осадков в этот же день.	Практика
Задача 7. Выбираем место под новый жилой комплекс	Построение линейчатых графиков	Лекция
	Вблизи развлекательного комплекса должны находиться густонаселенные жилые районы с доступными транспортными развязками. По карте, которая приведена в задаче, найдите территориальный пункт, от которого идет больше всего маршрутов к развлекательному центру.	Практика
Задача 8. Время работы развлекательного центра	Проанализируйте заданные графики. Определите оптимальный временной интервал работы центра, исходя из числа потенциальных посетителей и расходов на обслуживание центра отдыха.	Практика
Задача 9. Сколько "счастья" надо?	Построение круговых-диаграмм	Лекция
	В рамках данной задачи необходимо определить оптимальное количество возможных предприятий фудкорта, развлечений и торговли.	Практика
Задача 10. Конкурс заявок	Список заявок на открытие своих представительств в центре отдыха больше, чем установленное в задаче 9 рекомендуемое количество. Надо решить, чьи заявки лишние.	Практика



	Пользуясь графиком интересов жителей города и опираясь на мнение горожан, постройте список предприятий, заявки от которых не будут реализованы	
Модуль 2. Постановка проблемы	Необходимо провести исследования городского парка Энска "Роднички". Место отдыха "Роднички" расположено на территории прекрасного природного массива, но, к сожалению, по прошествии многих лет концепция парка устарела: его аллеи опустели, он больше не является сердцем города. Пришло время нового содержания для этого замечательного места.	Лекция
Задача 1. Ищем аналоги	Один из вариантов модернизации парка — это поиск готовых решений среди парков других городов. Необходимо, используя данные из заданной таблицы, найти 3 города с наиболее схожими с Энскими погодными условиями.	Практика
Задача 2. Единицы измерения площадей земельных участков	Составьте программу, позволяющую осуществлять перевод Га в кв.км. и наоборот (кв.км. в Га).	Практика
Задача 3. Парковые зоны	На основе проекта и статистики необходимо определить возможные размеры всех зон парка в кв.метрах. Общая парковая зона 15 гектаров (15 Га).	Практика
Задача 4. Замечательные места	Написать программу для поиска наиболее посещаемых зон в парках исследуемых городов. Необходимо найти 5 таких зон.	Практика
Задача 5. "Фишки" в облаках	У вас есть пять текстов — обобщённых записей, опубликованных в социальных сетях пяти "замечательных мест".	Практика

	К каждому из них постройте облако слов.	
Задача 6. Предпочтения	По заданному графику предпочтений горожан выберите два объекта, которые стоит добавить в проект при реконструкции парка.	Практика
Задача 7. Удобство в дорогах	Дана карта-схемы территории парка. Каждая точка карты-схемы проектируется как место для размещения определенного паркового объекта. Парковый объект будет посещаемым, если точка является доступной. Необходимо найти все такие точки.	Практика
Задача 8. Центр.	Центрированное место — это точка, из которой до других точек выходит наибольшее количество быстрых маршрутов (таких маршрутов, в которых все точки удовлетворяют условию доступности). Иначе, от центра любой объект находится в доступной точке. Используя схему парка, найти такой центр.	Практика
Задача 9. Центральная достопримечательность	Используя результаты опроса людей, построить график усредненных значений оценки достопримечательностей по данным из файла. Укажите достопримечательность с самым высоким баллом.	Практика
Задача 10. Время работы	Используя три графика активности для разных возрастных категорий, определите оптимальное время работы парка.	Практика
Модуль 3. Постановки проблемы.	Исследование социальных активностей, поведения поможет выявить потребности жителей района, их предпочтения, укажет на проблемы.	Лекция
	Парсинг данных	Лекция

<p>Задача 1. Сбор данных для исследований</p>	<p>Дан сайт, на котором собраны отзывы о знаковых местах района Эн. Необходимо написать программу-парсер, который соберет данных с интернет-ресурса. Для этого нужно разобраться с кодом страницы сайта, найти в коде "метки" (теги), которые будут указывать парсеру, какие данные и откуда считывать.</p>	<p>Практика</p>
<p>Задача 2. Анализ отзывов</p>	<p>Необходимо обработать обобщенный текст из Задачи 1, выделить в нем ключевое слово, то которое чаще всего используют пользователи, описывая исследуемый район Энска. Сгенерировать облако слов по всем отзывам</p>	<p>Практика</p>
<p>Задача 3. Анализ целевой аудитории района</p>	<p>Задачи на работу с массивами и списками.</p>	<p>Лекция</p>
	<p>Администрацией района Эн проводился опрос населения. Вам необходимо проанализировать данные и построить диаграмму-пирог с заданными категориями. Определите, какова доля участников каждой категории в опросе. Расположите группы в порядке убывания их количества: от самой большой группы к самой маленькой.</p>	<p>Практика</p>
<p>Задача 4. Какие в районе проблемы?</p>	<p>Вам предоставляются данные, представляющие из себя список проблем данного района, выявленных в результате опросов, интервью и данных с интернет-ресурса города. Вам необходимо найти 2 главные проблемы территории.</p>	<p>Практика</p>
<p>Задача 5. Формирование рейтинга мест</p>	<p>Рейтинг выстраивается по количеству лайков, жалоб, по посещаемости места. Воспользуйтесь заданным</p>	<p>Практика</p>

	графиком показателей мести сформируйте рейтинг. Укажите топовое и худшее место, по версии пользователей.	
Задача 6. Что важнее?	Администрация района предложила определить жителям приоритет финансового вложения в решение проблем города. Можно выделить одну проблему и потратить выделенные финансы только на одно решение, либо выбрать две с распределением финансов. Используя данные опроса укажите, сколько процентов жителей района предложили решать обе проблемы одновременно?	Практика
Задача 7. Умные решения	Определите оптимальное количество умных решений достаточных для обеспечения комфортной среды обитания района и укладывающихся в бюджет.	Практика
Задача 8. Плотность предприятий в районе	Исходя из выявленных проблем и определенных умных решений, нужно порекомендовать администрации, в каком районе лучше открыть заведение общественного питания	Практика
Задача 9. Где открыть ресторан?	Найти район города, в котором открытие нового предприятия общественного питания, например ресторана, будет наиболее целесообразным.	Практика
Задача 10. Ресторан или кафе быстрого питания?	Используя данные двух списков, необходимо определить, какого типа будет предприятие: кафе быстрого питания, ресторан или кафе.	Практика
Раздел 2. Командное решение задач		

<p>Модуль 3. Знакомство с Легендой</p>	<p>Мэр и администрация Энска стараются создавать в городе наиболее благоприятные условия для жизни. Вам необходимо провести исследование, результатом которого будет изучение настроения жителей различных кварталов города и рекомендации по проектированию или реконструкции объектов городской инфраструктуры.</p>	<p>Лекция</p>
<p>Задача 1. Получение данных в Google Collab.</p>	<p>Создание тетрадки, настройки доступа к тетрадке для членов команды и экспертов. Разметка тетрадки текстовыми блоками. Правила безопасности при работе с тетрадкой</p>	<p>Практика</p>
<p>Задача 2. Анализ статьи мэра</p>	<p>Проанализируйте обращение мэра на сайте Дистрикта, выявите по данным мест какая именно проблема существует в районе из описанных в выступлении мэра. Сформулируйте несколько предположений(гипотез), с чем связана данная проблема, на основе информации, предоставленной на сайте данных.</p>	<p>Практика</p>
<p>Задача 3-4. Парсинг сайта для извлечения данных.</p>	<p>Вам предлагается воспользоваться инструментами парсинга: извлечь информацию из содержимого элемента html или значений атрибутов и сделать преобразование типов в соответствии с типом информации.</p>	<p>Практика</p>
<p>Задача 5. Анализ зеленых насаждений городской среды</p>	<p>На основе фотографии района с квадрокоптера определить отношение зеленых насаждений к другим объектам. Для анализа изображения воспользуйтесь сервисом по определению цветов с картинки (например: <a href="https://hysy.org/get_colors">https://hysy.org/get_colors</a>)</p>	<p>Практика</p>

<p>Задача 6. Анализ зеленых насаждений (нахождение аналога)</p>	<p>На основании представленных данных найдите город-аналог со схожим процентным содержанием зеленых насаждений по сравнению с вашим районом. В ответе укажите название найденного города.</p>	<p>Практика</p>
<p>Задача 7. Интересы Дистрикта</p>	<p>Проанализируйте интересы исследуемого района города N на основе данных, представленных в файле интересов Дистрикта. Найдите 5 типов интересов, которые чаще всего встречаются в Дистрикте. Постройте столбчатый график интересов.</p>	<p>Практика</p>
<p>Задача 8. Интересы Дистрикта (поиск аналога)</p>	<p>Проанализируйте данные в файлах по интересам трех других городов. Составьте рейтинг из 5 приоритетных интересов жителей. Сравните результаты интересов вашего Дистрикта и получившиеся топы интересов трех других городов. Определите город(город-аналог), наиболее похожий по интересам на ваш Дистрикт.</p>	<p>Практика</p>
<p>Задача 9. Публикация аналитических выводов</p>	<p>На основе полученного аналитиком команды графика интересов вашего Дистрикта сделайте выводы по интересам жителей вашего района и опубликуйте их в тетради вместе с графиком.</p>	<p>Практика</p>
<p>Задача 10-11. Визуализация мест</p>	<p>Работа с картами</p>	<p>Лекция</p>
	<p>Создайте визуализацию всех точек района по их координатам на карте, такие точки называют геометками. При визуализации геометок отобразите популярность и тип каждой метки. Разместите получившуюся карту Дистрикта в эссе, сделайте выводы о корректности расстановки геометок.</p>	<p>Практика</p>

<p>Задача 12 Частотный анализ отзывов.</p>	<p>Сгенерируйте облако слов по всем отзывам, чтобы распределить слова отзывов по частоте их упоминания. Исключите такие слова как: место, здесь, очень. Возьмите для анализа отзывов только слова длиной от 4 букв и больше. Определите ключевое существительное, которое пользователи используют чаще всего.</p>	<p>Практика</p>
<p>Задача 13. Семантический анализ отзывов</p>	<p>Отзывы имеют оценку пользователей. Считается, что рейтинг от 3 и ниже определяет отзыв как негативный, а позитивные имеют оценку 4 или 5. Распределите отзывы пользователей с сайта дистрикта города N по настроению. Постройте облака слов к каждому из настроений, чтобы найти главную позитивную сущность и негативную.</p>	<p>Практика</p>
<p>Задача 14. Анализ отзывов (завершение)</p>	<p>По результатам анализа отзывов добавьте в эссе получившиеся облака слов и выводы, которые по итогу решения данных задач можно сформулировать</p>	<p>Практика</p>
<p>Задача 15-16. Посещаемость мест</p>	<p>На основе данных посещаемости мест с сайта необходимо найти самое посещаемое место в вашем Дистрикте. Постройте столбчатую диаграмму по среднему значению посещаемости мест в Дистрикте. Найдите самое посещаемое место в вашем Дистрикте города N. Оформите результаты анализа посещаемости мест Дистрикта в эссе</p>	<p>Практика</p>
<p>Задача 17. Центрированная карта</p>	<p>Определите центрированную точку(и), исходя, из условной карты дорог между точками района(графа) представленной в задаче. При этом перекресток точкой не считается. Определите</p>	<p>Практика</p>

	номер самой центрированной точки по району, если их несколько, то укажите одну(любую) из них.	
Задача 18. Непопулярность точки	Сравните данные посещаемости центрированной точки и самого посещаемого места. Сделайте соответствующие выводы. Сделайте предположения почему центрированная точка может оказаться не самой посещаемой. Опубликуйте свои выводы в эссе. Вычислите у центрированной точки максимальное число посещений за день.	Практика
Задача 19. Плотность района	Рассчитайте плотность Дистрикта города N по данным плотности района. Проанализируйте данные по плотности Дистрикта.	Практика
Задача 20. Поиск аналогичного по плотности района	Проанализируйте данные о средней плотности из районов трех других городов и плотности вашего Дистрикта. Определите город аналог для вашего Дистрикта	Практика
Задача 21. Посещаемость города-аналога	Задание программиста: построить столбчатую диаграмму по показателю о средней посещаемости мест Дистрикта. Задание аналитика: взять данные из задачи 15, а данные по текущему шагу взять из результатов работы программиста команды по этому же заданию. Сравните полученные результаты по вашему Дистрикту из двух источников и сделайте выводы по точкам с низкой посещаемостью и максимальной. Оформите результаты сравнения в итоговом эссе.	Практика
Задача 22. Настроение города	Напишите функцию, которая по условию, указанному в условии	Практика



	задачи, расшифровывает цветовой код улицы	
Задача 23-24. Исследование климата	NumPy: сложные задачи	Лекция
	Вычислите среднее значение температуры воздуха анализируемого вами района города N по представленным данным. По полученному среднему значению температуры Дистрикта найдите его "климатический" двойник среди данных о районах других городов. Внесите в итоговую публикацию получившиеся ранее значение средней температуры воздуха исследуемого Дистрикта	Практика
Задача 25. Настроение города	По номеру города аналога, возьмите координату точки. Отыщите эту точку на Google Картах. С помощью сервиса определения главных цветов определите три ключевых цвета улицы. С помощью созданного в задаче 22 кода расшифруйте полученный цветовой код. Сделайте вывод о том, имеет ли смысл скопировать этот цветовой код в вашем Дистрикте.	Практика
Задача 26. Центрированная карта	Соотнесите точки из заданного графа с точками на графической карте с геометками вашего района, получившейся на шаге 18. И отобразите все дороги между инфраструктурными точками. Разместите новую карту района в своем эссе.	Практика
Задача 27. Добавление дорог	Модифицируйте карту Дистрикта таким образом, чтобы самая посещаемая точка стала самой центрированной. Для этого вы можете добавить новые дороги между пунктами.	Практика

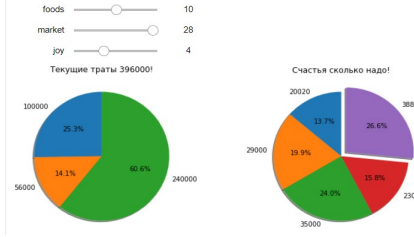
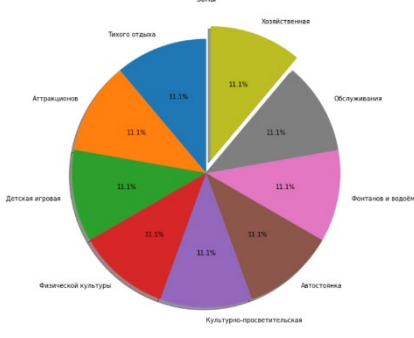
<p>Задание 28. Сопоставление негативных и позитивных отзывов</p>	<p>Проанализируйте количество негативных и позитивных отзывов в Дистрикте. Составьте диаграмму-пирог по анализу количества негативных и позитивных отзывов и разместите ее в своем эссе. По итогу анализа укажите количество негативных и количество позитивных отзывов.</p>	<p>Практика</p>
<p>Задание 29. Улучшение системы дорог</p>	<p>Обработка графиков в графическом редакторе.</p>	<p>Лекция</p>
	<p>Нанесите на созданную вами карту с геометками и дорогами дополнительные дороги, которые спроектировали на шаге 27. Выделите их цветом или формой.</p>	<p>Практика</p>
<p>Задание 30. Оформление вычислительного эссе</p>	<p>В результате выполнения командой заданий появилось несколько графических объектов. Добавьте на них подписи, визуализируйте ответ при помощи выделения нужно столбца или сектора на них, установите на графике метки</p>	<p>Практика</p>
<p>Задание 31. Новые объекты городской инфраструктуры</p>	<p>Исходя из полученных данных в ходе анализа по Дистрикту города N (погода, плотность, его текущая инфраструктура и т. п.), выбрать места куда лучше всего установить новые объекты городской инфраструктуры. Обосновать потребность в новых объектах не только логически, но и результатами анализа данных по району. Показать, как может меняться настроение города в зависимости от места и положения установленного вами объекта. Нанести новые объекты на итоговую карту Дистрикта. Оформить все выводы в эссе.</p>	<p>Практика</p>
<p>Задание 33. Манипуляторы</p>	<p>Построение интерактивных диаграмм с манипуляторами</p>	<p>Лекция</p>

	Для того, чтобы территория процветала, и среда обитания населения города была комфортной, а люди счастливыми, необходимо наделить район объектами в соответствии с интересами населения. Определите оптимальное количество умных решений достаточных для обеспечения комфортной среды обитания района и укладываемых в бюджет.	Практика
--	--	----------

## Контрольно-измерительные материалы курса

В первом разделе курса, во время выполнения индивидуальных заданий, оценка работы учащихся складывается из правильности выполнения задач. Ответы на задачи представлены в таблице:

Номер задачи	Правильный ответ
Модуль 1	
Задача 2. Нужны ли городу развлекательные центры?	центр отдыха с минимальной посещаемостью под буквой D
Задача 3. Как найти проходное место в городе?	89, 50 или 89, 49
Задача 4. Маршруты	2 маршрута, 5 км минимальное расстояние
Задача 5. Анализ сообщений в социальных сетях	СТРОЙКУ, ДОЖДИ, СТРЕЛА
Задача 6. Погода в городе	преобладающая погода пасмурная. Количество дней и тип погоды Пасмурная 182 Плохая 73 Проголочная 73 Хорошая 37
Задача 7. Выбираем место под новый жилой комплекс	Наибольшее количество маршрутов идет из пункта B
Задача 8. Время работы развлекательного центра	оптимальный режим работы центра с 10 утра до 18 вечера

Задача 9. Сколько "счастья" надо?	
Задача 10. Конкурс заявок	Нужно отклонить заявку ресторана "Две Шпаги" и заявку Облачного парка
Модуль 2.	
Задача 1. Ищем аналоги	города под номерами 3,9 и 28 наиболее схожи с Энском
Задача 3. Парковые зоны	
Задача 4. Замечательные места	[2,31], [5,4], [10,12], [35,49], [48,30]
Задача 5. "Фишки" в облаках	Кораблики Барабаны дождя Песочники Forrest Cafe Музей
Задача 6. Предпочтения	<p>Стоит добавить в проект:</p> <ul style="list-style-type: none"> <li>• Детский парк под открытым небом (но застекленный).</li> <li>• Колесо обозрения. Окна в каютах при соприкосновении стекла с каплями дождя небо в окнах становится радужными.</li> </ul>
Задача 7. Удобство в дорогах	Точки Д и Л располагаются за пределами доступности
Задача 8. Центр.	Центр исследуемого объекта - точка Б
Задача 9. Центральная достопримечательность	Колесо обозрения
Задача 10. Время работы	Оптимальное время работы с 9 утра до 17 вечера
Модуль 3.	
Задача 1. Сбор данных для исследований	название места: h2, описание к фото: р.
Задача 2. Анализ отзывов	чаще всего упоминается слово "Природа"

Задача 3. Анализ целевой аудитории района	Родители с детьми, Другое, Молодежь и Пенсионеры
Задача 4. Какие в районе проблемы?	Мусор и Скучно
Задача 5. Формирование рейтинга мест	Худшее место: Набережная Лучшее место: Парк Динамо
Задача 6. Что важнее?	60%
Задача 7. Умные решения	возможный вариант ответа 5 18 2 1
Задача 8. Плотность предприятий в районе	0.0033, 0.0016, 0.0008
Задача 9. Где открыть ресторан?	3,37355
Задача 10. Ресторан или кафе быстрого питания?	5775.3

Во втором разделе курса система оценивания несколько усложняется. На рисунке ниже приведена схема структуры системы оценивания. На схеме структуры оценивания можно видеть типы оценки – автоматическая и экспертная, типы задач для автоматической оценки, основные критерии экспертной оценки.

Система оценивания разбита на две части: автоматизированную оценку, через систему выполнения тестовых заданий, и экспертную оценку - с участием четырех внешних (приглашенных) экспертов. При этом вклад в оценку автоматической части составляет 60%, экспертной 40%. Кроме того, за одну из задач (задачу, которую участники сами ставят и сами предлагают оригинальное решение) судьи могут начислить дополнительно до 10 баллов. Структура оценки также делится на выполнение отдельных заданий соревнований, а также создание командой и демонстрацию (защиту) комплексного документа – публикации результатов исследований в документе Google Collab.

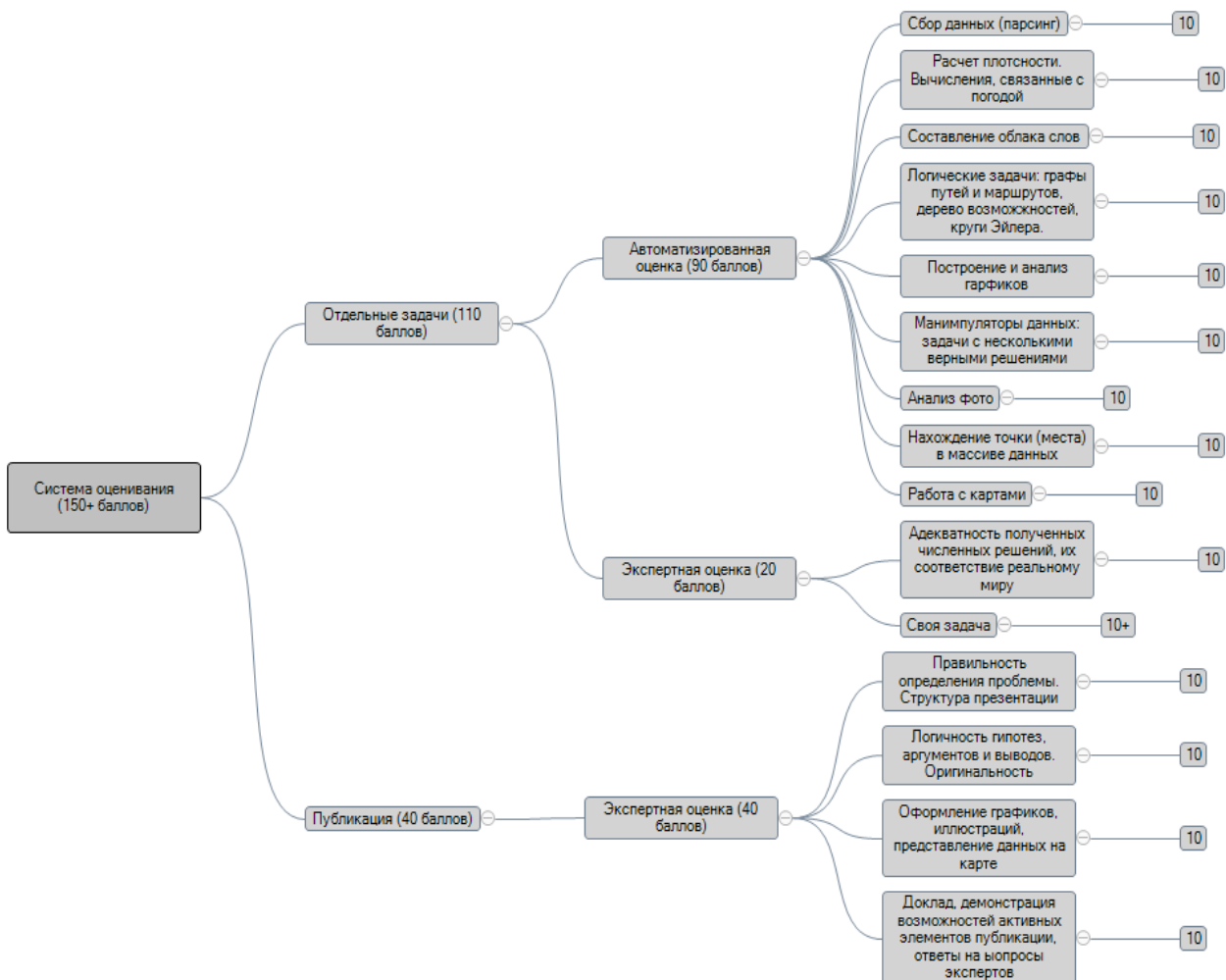
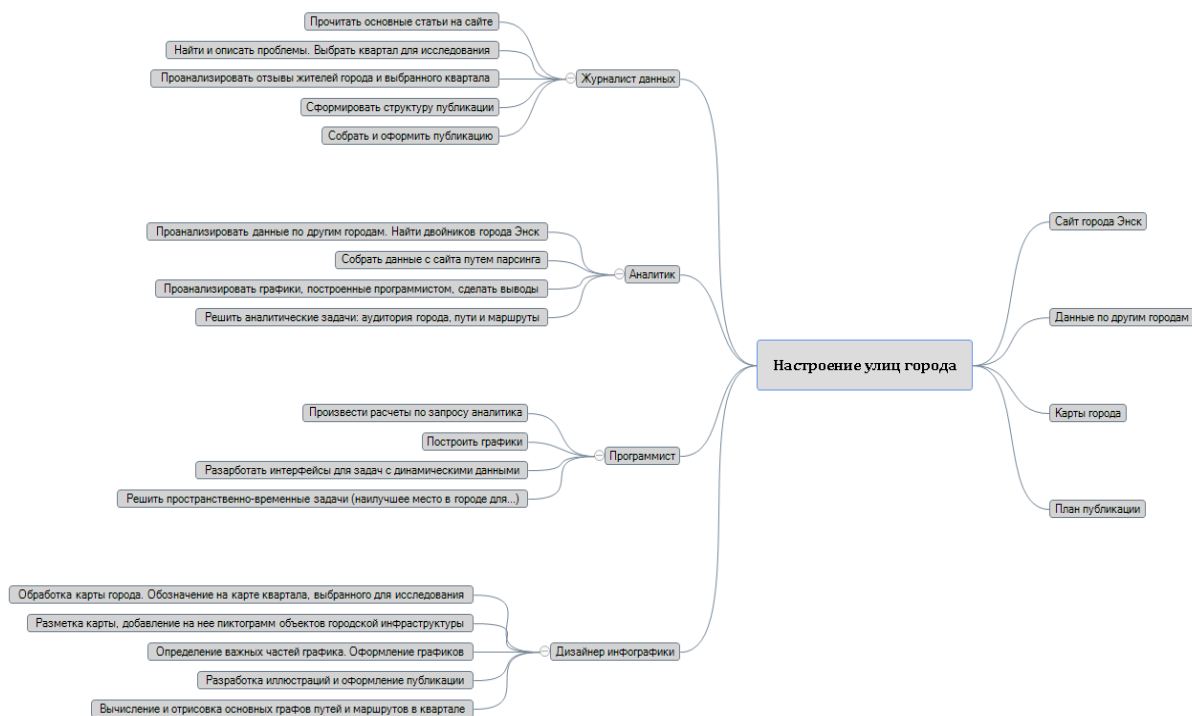


Рис. Схема структуры системы оценивания участников финала

Структура заданий командного этапа с условным разделением по ролям, представлена на рисунке ниже.

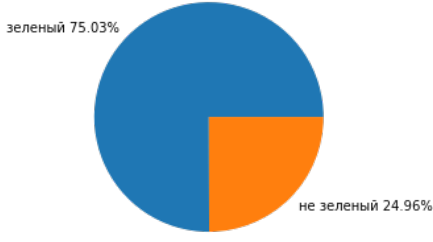
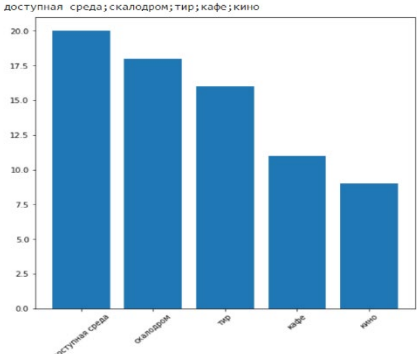


*Структура заданий командного этапа с условным разделением по ролям (слева) и исходными данными (справа)*

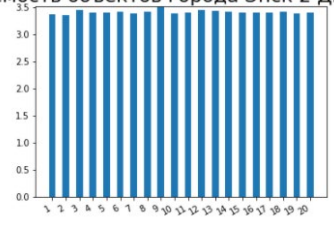
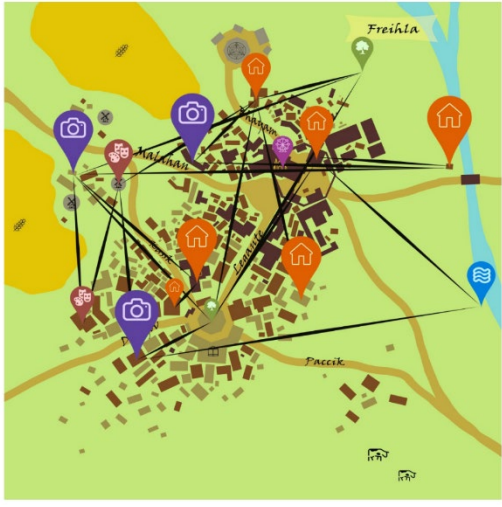

Команда участников, набравшая наибольшее количество баллов, становится победителем. Определенные типы задач могут представлять для участников трудности, в этом случае участники в праве заменить эти задания на те, которые они умеют решать хорошо. Здесь хотелось бы уточнить, что за правильно решенные однотипные задачи баллы снижаются. С максимальной оценкой допускается решение двух задач, за каждую последующую задачу одного и того же типа начисляется меньше на два балла: 1 и 2 задачи- 10 баллов, 3 задача -8, 4 -7, 5 -6 и т.д.

#### Ответы на задачи раздела 2.

Номер задачи	Правильный ответ
Задача 2. Анализ статьи мэра	Дистрикт 2
Задача 3-4. Парсинг сайта для извлечения данных.	img,name,category,alt

<p>Задача 5. Анализ зеленых насаждений городской среды</p>	<p>District 2</p>  <p>зеленый 75.03%</p> <p>не зеленый 24.96%</p>
<p>Задача 6. Анализ зеленых насаждений (нахождение аналога)</p>	<p>Уфа</p>
<p>Задача 7-8. Интересы Дистрикта</p>	 <p>доступная среда; скейтпарк; тир; кафе; кино</p> <p>доступная среда; тир; скейтпарк</p>
<p>Задача 9. Публикация аналитических выводов</p>	<p>Самым популярным интересом является “доступная среда”</p>
<p>Задача 10-11. Визуализация мест</p>	<p>жилой комплекс-ласточка ЦПКиО имеет самую маленькую геометку, хотя его местоположение достаточно хорошее</p>
<p>Задача 12 Частотный анализ отзывов.</p>	<p>площадка</p>
<p>Задача 13. Семантический анализ отзывов</p>	<p>парк; площадка</p>
<p>Задача 15-16. Посещаемость мест</p>	<p>Галерея Детроит</p>
<p>Задача 17. Центрированная карта</p>	<p>13</p>
<p>Задача 18. Непопулярность точки</p>	<p>7</p>
<p>Задача 19. Плотность района</p>	<p>5,3</p>
<p>Задача 20. Поиск аналогичного по плотности района</p>	<p>3.40-3.40</p>



<p>Задача 21. Посещаемость города-аналога</p>	<p>Средняя посещаемость объектов города Энск 2 Дистрикта за 20 дней</p> 
<p>Задача 22. Настроение города</p>	<p>28.67;27.57</p>
<p>Задача 23-24. Исследование климата</p>	<p>средняя температура воздуха в анализируемом Дистрикте составляет 23.23.</p>
<p>Задача 25. Настроение города</p>	<p>В палитре города аналога преобладает серый и его оттенки</p>
<p>Задача 26. Центрированная карта</p>	
<p>Задача 27. Добавление дорог</p>	
<p>Задание 28. Сопоставление негативных и позитивных отзывов</p>	<p>12,33</p>
<p>Задание 33. Манипуляторы</p>	<p>2 10 2 3 3 3 14 13 1</p>

Критерии оценки, которыми должны руководствоваться члены жюри приведены в таблице ниже.

*Таблица. Критерии оценивания выступлений команд на итоговом занятии этапе*

<b>Критерии (максимальный балл)</b>	<b>Итого баллов</b>
Критерий 1. Правильность определения проблемы. Структура публикации. Логичность гипотез, аргументов и выводов. Оригинальность найденных решений (содержание)	Максимум 9
Проблема выделена правильно, предложены гипотезы возникновения проблемы	2 балла
Публикация оформлена в структуру: есть заголовки и содержание, вычисления и графики, указаны авторы публикации	2 балла
Выводы к задачам анализа данных	Сделаны более чем 60% задач - 3 балла Сделаны к некоторым задачам - 1 балл
Сделан общий вывод в публикации	1 балл
Предложены оригинальные решения для проблем	1 балл
Критерий 2. Оригинальность найденных решений (технология). Выбор методов, используемых для решения задач (программирование). Оценка наличия интерактивных элементов	Максимум 10 баллов
При решении задач использовали программирование	в основном - 2 балла иногда -1 балл не использовали – 0 баллов
Интерактивная карта	3 балла
Графики с манипуляторами	2 балла
Парсинг данных выполнен автоматически	3 балла
Критерий 3. Оформление графиков, иллюстраций, представление данных на карте, оформление графа дорог	Максимум 10 баллов
Карта с нанесением на нее меток категорий заведений, отличающихся размером в зависимости от количества лайков у заведения	4 балла
Графики и их оформление	оформлен выделением цветом ответа - 2 балла есть подписи на шкалах 1 есть иллюстрация к графику -1

На карту нанесен граф дорог	граф есть -1 балл дорога выделена цветом или формой - 1 балл
Критерий 4. Доклад, демонстрация возможностей активных элементов публикации, ответы на вопросы экспертов, соблюдение регламента	Максимум 11 баллов
Доклад и публикация (эссе)	Доклад публикации представлен, как единое целое есть выводы – 3 балла как отдельные элементы, есть выводы – 2 балла элементы доклада не связаны совсем, выводы в конце доклада не сделаны – 1 балл
В тетради есть активное содержимое (манипуляторы, карта) ребята демонстрируют, как работает это механизм	2 балла
Графики в публикации	5 и более графиков и (или) иллюстраций – 3 балла не более 4 графиков и (или) иллюстраций – 2 2 и менее – 1 балл
Представитель команды отвечает на вопросы экспертов уверенно	очень – 2 балла владеет информацией слабо - 1 балл не владеет информацией – 0 баллов
Соблюден регламент доклада	1 балл

### Материально-техническое оснащение кружка

- Персональный компьютер или ноутбук с выходом в сеть Интернет для каждого учащегося
- Графический планшет (особенно, для учащихся ролью дизайнер).
- Установленное на компьютере программное обеспечение:
  - Excel,
  - Python 3,
  - Paint (Inscapе, Gimp, любой другой графический редактор)

## Методические рекомендации к проведению первого раздела программы

При составлении задач данного раздела учитываются особенности подросткового возраста, а именно:

- Стремление к реализации своих способностей
- Интерес к социальным проблемам
- Осознание себя в обществе
- Стремление к признанию своих заслуг.

Тема трека подразумевает целый комплекс возможных социальных проблем, связанных с непосредственным окружением школьников. Содержание задач и методы их решения, выходящие за рамки привычного школьного образования помогают осознанию своих интересов. Участие в соревнованиях питает стремление к признанию заслуг.

Также следует отметить, что задачи как первого раздела, так и второго представляют собой единую неразрывную историю, рассказанную в традициях сторителлинга и, решая задачи, участники дописывают эту историю, принимают в ней непосредственное участие, а затем рассказывают собственную.

При построении задач соблюдаются принципы «от простого к сложному», «от конкретного к абстрактному», что позволяет даже слабоподготовленному школьнику принять участие в отборочном туре и проявить себя в одной из ролей.

Для того, чтобы выстроить историю через серию задач требуется преамбула, введение. Ниже представлено введение, которое можно использовать на первом занятии для учащихся.

## Методические рекомендации по проведению второго раздела программы.

В данном разделе программы формируются команды для выполнения заданий.

На этапе подготовки к итоговому заданию учащиеся знакомятся с понятием вычислительного эссе. Это знакомство проходит в три этапа. Первый этап связан с рассмотрением готового вычислительного эссе: ребята получают доступ к интерактивной публикации, в которой смогут менять данные, анализировать их, делать выводы, строить графики. На втором этапе ребята работают с вычислительным эссе, в котором есть данные, а функции для их обработки и визуализации надо разработать самостоятельно. На третьем этапе учащиеся получают задание для самостоятельной разработки вычислительного эссе.

Дополнительно нужно провести мастер-классы по инфографике и написанию историй, основанных на данных (сторителлинг).

Для оценки итогового задания рекомендуется использование автоматической (компьютерной) оценки для критериев, связанных с корректностью вычислений, а также подсчетом реализаций определенных функций в вычислительном эссе. Для оценки творческого компонента интерактивной публикации можно привлечь экспертов из числа сотрудников технологических партнеров сферы.

## Методические материалы курса

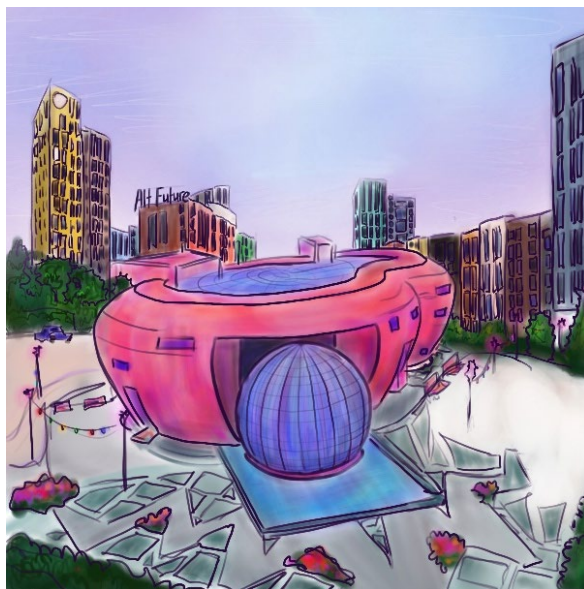
### Первый модуль индивидуального этапа

#### Постановка проблемы для участников.

Отдел «неЛинейного счастья» компании AltFuture ведет работу по исследованию условий создания комфортной городской среды обитания.

Градоначальники, заботящиеся о достойном жизнеобеспечении своих граждан, стали обращаться в новый отдел за советом и анализом ситуации.

Однажды в отдел «нелинейного счастья» пришел мэр города Энска. Он рассказал, что совсем недавно, достроил в Энске новый развлекательный центр с парком аттракционов, и надеялся, что это место будет центром отдыха для горожан и гостей города (рис. 1).



*Рис. 1. Торговый центр г.Энск*

Полученные результаты послужат разработке проекта по улучшению качества инфраструктуры территории за счет ее трансформации, а в городе появится еще один район с комфортной средой обитания.

### **Задача 1. Что такое куча?**

#### **Условие.**

За какую работу ты бы не взялся, необходимо проанализировать все аспекты. Помните, мультфильм про удава, слоненка, обезьянку и попугая? Был в этом мультфильме разговор о том, что считать кучей. Так же и у нас 5 человек — это куча людей? Ну да, а если их надо уместить на площади 0,5 кв.м? Мало людей в новом центре отдыха — это понятие относительное. Если людей в городе в принципе не так много, например, всего 2000, а территория города

очень большая, нет ничего удивительного, что в среднем для некоторого отдельно взятого территориального района плотность горожан невелика.

Потому, первое что необходимо выполнить стажерам, написать программу, которая будет определять количество людей на квадратный метр — это покажет, сколько в среднем людей мы можем ожидать в центре отдыха с определенной площадью.

На вход дано два числа:  $N$  - количество людей в городе и  $S$  - площадь города в метрах. На выходе нужно получить количество людей на квадратный метр, округленное по правилам математики.

### Пример данных

ВХОД: 16 2

ВЫХОД: 8

### Решение.

Пример решения на Python 3

```
st = input().split(" ")
result = int(st[0])/int(st[1])
if result>0:
    result = int(result+0.5)
else:
    result = int(result-0.5)
print(result)
```

## Задача 2. Нужны ли городу развлекательные центры?

### Условие

Также возможно, что у горожан нет потребности в центре отдыха - они вообще не посещают развлекательные центры. Необходимость в уточнении этого факта, ставит перед стажерами новую задачу.

В таблице представлено количество людей - посетителей развлекательных центров в разных районах города в период с января по декабрь 2019 года. Задача стажеров- построить столбчатую диаграмму и ответить на

вопрос, какой буквой закодирован торговый центр с наименьшей посещаемостью.

Центр отдыха с минимальной посещаемостью - исследуемый развлекательный центр. В таблице посещаемость указана количеством человек на 100 кв. метров. Посещаемость 30 000 человек в год на 100 кв.метров - считается нормой.

*Таблица 2. Пример данных о посещаемости*

	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>
<b>2019-01-01</b>	83	98	113	46	9
<b>2019-01-02</b>	67	80	109	48	63
<b>2019-01-03</b>	88	122	106	96	34
<b>2019-01-04</b>	80	103	125	45	35
<b>2019-01-05</b>	92	110	111	65	30
...	...	...	...	...	...
<b>2019-12-27</b>	63	83	136	98	59
<b>2019-12-28</b>	76	89	102	41	33
<b>2019-12-29</b>	62	114	115	51	50
<b>2019-12-30</b>	68	107	125	57	49
<b>2019-12-31</b>	98	120	124	44	33



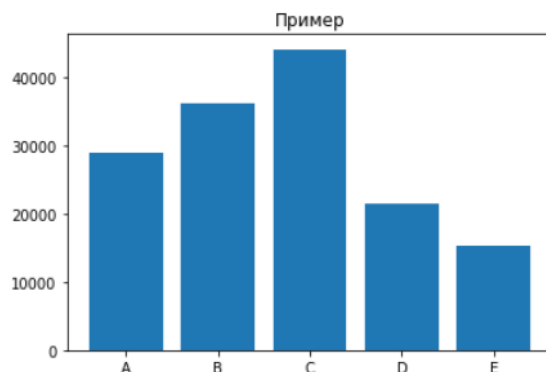


Рис. 2. Посещаемость центров отдыха

В данном случае центр отдыха с минимальной посещаемостью обозначен буквой Е.

**Входные данные для решения задачи:**

A = [85, 102, 83, 93, 106, 63, 83, 118, 95, 130, 110, 95, 70, 140, 85, 87, 96, 85, 98, 69, 122, 88, 95, 96, 93, 80, 102, 92, 106, 75, 95, 103, 92, 75, 86, 71, 87, 108, 154, 91, 125, 116, 87, 109, 93, 82, 100, 87, 100, 112, 76, 116, 115, 93, 98, 95, 96, 92, 108, 106, 121, 97, 134, 85, 92, 76, 100, 94, 92, 99, 104, 116, 100, 81, 86, 98, 85, 114, 84, 97, 76, 82, 134, 105, 73, 101, 98, 109, 99, 114, 124, 95, 94, 111, 96, 72, 87, 90, 94, 102, 109, 86, 109, 84, 85, 123, 82, 99, 85, 106, 121, 112, 95, 117, 79, 118, 110, 117, 93, 94, 113, 93, 110, 105, 97, 101, 126, 96, 96, 128, 67, 101, 96, 99, 95, 107, 112, 111, 108, 95, 120, 84, 88, 104, 110, 113, 119, 81, 105, 89, 116, 91, 90, 108, 96, 114, 102, 94, 117, 101, 88, 102, 103, 87, 85, 101, 98, 127, 80, 111, 84, 88, 114, 94, 110, 95, 92, 102, 124, 113, 94, 102, 102, 90, 96, 121, 74, 106, 91, 88, 96, 103, 94, 107, 87, 114, 102, 79, 79, 98, 110, 90, 100, 101, 95, 110, 132, 88, 119, 75, 114, 72, 73, 87, 85, 97, 85, 116, 96, 85, 126, 95, 89, 96, 79, 98, 118, 96, 126, 91, 113, 90, 81, 122, 81, 80, 108, 109, 92, 113, 116, 99, 103, 112, 118, 113, 86, 84, 117, 101, 94, 107, 88, 84, 115, 117, 109, 106, 107, 86, 123, 114, 80, 75, 86, 69, 101, 59, 73, 88, 96, 99, 112, 105, 100, 125, 90, 107, 100, 91, 96, 96, 116, 115, 129, 98, 95, 96, 87, 107, 98, 99, 104, 71, 107, 113, 105, 94, 110, 111, 103, 110, 100, 121, 93, 119, 113, 103, 111, 90, 84, 96, 116, 105, 89, 123, 127, 100, 94, 111, 81, 90, 135, 115, 90, 101, 90, 72, 101, 104, 92, 94, 94, 90, 99, 109, 105, 102, 94, 86, 120, 102, 110, 127, 90, 94, 109, 118, 106, 98, 94, 107, 70, 88, 89, 107, 87, 102, 84, 106, 116, 96, 91, 109, 103]

B = [107, 87, 93, 42, 111, 88, 69, 96, 57, 65, 109, 86, 60, 59, 68, 81, 78, 72, 57, 94, 86, 98, 63, 81, 98, 74, 108, 76, 81, 120, 104, 68, 90, 85, 67, 68, 70, 75, 106, 65, 82, 67, 97, 125, 98, 95, 77, 93, 104, 68, 84, 89, 107, 79, 121, 65, 94, 85, 103, 102, 67, 76, 53, 84, 75, 53, 72, 74, 98, 79, 102, 68, 46, 91, 64, 52, 70, 95, 114, 70, 142, 69, 70, 120, 88, 82, 91, 95, 89, 74, 89, 98, 52, 77, 85, 106, 49, 104, 77, 93, 65, 74, 65, 108,

66, 119, 56, 91, 32, 71, 64, 73, 129, 106, 103, 60, 72, 96, 87, 91, 104, 80, 91, 82, 78, 58, 76, 80, 67, 95, 73, 74, 59, 72, 52, 71, 98, 127, 64, 78, 77, 67, 77, 99, 52, 93, 110, 102, 50, 62, 96, 56, 110, 46, 62, 71, 113, 72, 112, 71, 80, 68, 84, 88, 58, 97, 57, 108, 57, 66, 89, 112, 47, 62, 52, 86, 93, 69, 48, 64, 61, 99, 73, 64, 94, 50, 87, 66, 119, 60, 48, 67, 83, 64, 79, 63, 72, 101, 85, 94, 74, 84, 89, 76, 60, 135, 74, 78, 107, 88, 68, 75, 92, 85, 88, 85, 74, 110, 102, 113, 80, 67, 66, 118, 86, 108, 69, 131, 46, 128, 63, 57, 114, 93, 84, 78, 62, 110, 91, 51, 51, 93, 95, 93, 51, 82, 64, 84, 75, 59, 67, 42, 87, 82, 101, 76, 73, 67, 96, 85, 23, 107, 40, 92, 85, 101, 78, 34, 98, 82, 76, 123, 32, 71, 56, 92, 71, 84, 76, 56, 77, 105, 77, 87, 75, 68, 74, 43, 72, 48, 84, 93, 79, 78, 62, 73, 85, 95, 92, 109, 106, 79, 87, 103, 104, 61, 63, 101, 80, 58, 61, 86, 84, 48, 79, 104, 99, 94, 110, 90, 87, 79, 45, 64, 75, 28, 87, 96, 94, 129, 105, 110, 97, 67, 58, 56, 40, 62, 88, 101, 73, 42, 52, 120, 108, 57, 69, 86, 83, 72, 71, 42, 119, 73, 67, 90, 64, 82, 72, 77, 80, 54, 89, 95, 60]

C = [160, 64, 123, 111, 157, 127, 82, 169, 121, 123, 126, 110, 147, 58, 149, 111, 187, 117, 71, 124, 111, 154, 200, 157, 68, 101, 112, 83, 188, 148, 122, 73, 98, 103, 143, 102, 126, 104, 144, 123, 98, 172, 137, 88, 106, 127, 173, 109, 98, 94, 111, 97, 104, 112, 128, 81, 123, 147, 87, 154, 139, 98, 117, 77, 78, 117, 129, 150, 91, 128, 106, 53, 150, 131, 144, 147, 132, 81, 125, 141, 150, 160, 156, 109, 128, 67, 121, 113, 78, 111, 147, 130, 127, 64, 117, 112, 42, 127, 166, 158, 131, 82, 171, 164, 58, 114, 164, 149, 111, 112, 150, 121, 116, 148, 129, 102, 125, 102, 69, 143, 160, 128, 179, 125, 178, 150, 109, 109, 127, 96, 92, 88, 115, 128, 126, 111, 121, 95, 158, 21, 116, 109, 67, 114, 116, 123, 141, 103, 91, 91, 98, 129, 81, 85, 114, 147, 115, 135, 116, 126, 121, 116, 121, 125, 158, 112, 140, 146, 122, 136, 178, 162, 123, 76, 178, 80, 103, 113, 68, 139, 173, 73, 89, 137, 123, 161, 135, 66, 128, 81, 108, 122, 148, 75, 118, 82, 50, 75, 84, 159, 91, 163, 127, 106, 88, 139, 78, 147, 108, 75, 135, 76, 109, 95, 123, 101, 129, 164, 149, 106, 82, 100, 98, 121, 89, 104, 142, 145, 115, 69, 127, 107, 145, 138, 210, 110, 123, 163, 158, 136, 123, 100, 141, 83, 126, 62, 131, 151, 129, 107, 165, 128, 144, 162, 82, 129, 164, 150, 88, 123, 92, 137, 167, 125, 103, 83, 81, 63, 127, 124, 98, 122, 102, 143, 123, 144, 80, 95, 154, 114, 62, 113, 107, 101, 104, 140, 113, 94, 144, 69, 125, 136, 111, 128, 136, 94, 118, 141, 97, 142, 115, 119, 138, 155, 53, 70, 137, 117, 128, 77, 150, 80, 128, 116, 111, 132, 117, 150, 173, 165, 140, 170, 121, 113, 136, 201, 107, 113, 92, 141, 114, 137, 159, 103, 70, 162, 92, 153, 145, 124, 74, 96, 83, 120, 65, 83, 170, 100, 108, 146, 134, 80, 98, 137, 124, 124, 87, 164, 71, 114, 114, 88, 115, 156, 114]

D = [60, 24, 49, 55, 45, 45, 49, 41, 60, 51, 51, 51, 39, 29, 44, 46, 44, 41, 50, 66, 50, 32, 58, 69, 50, 53, 55, 42, 38, 53, 40, 42, 56, 59, 62, 32, 52, 47, 49, 50, 36, 54, 59, 43, 37, 55, 44, 62, 42, 40, 32, 71, 46, 56, 54, 40, 43, 35, 40, 26, 53, 40, 41, 38, 44, 54, 54, 39, 40, 44, 46, 56, 76, 46, 53, 41, 41, 67, 48, 29, 49, 46, 41, 46, 53, 41, 48, 56, 39, 46, 47, 41, 63, 78, 49, 40, 51, 45, 50, 49, 67, 58, 62, 45, 56, 55, 32, 54, 34, 45, 46,

46, 45, 66, 49, 53, 57, 44, 55, 43, 32, 31, 43, 55, 47, 40, 50, 63, 53, 55, 41, 77, 49, 4  
6, 60, 53, 59, 54, 55, 47, 62, 42, 65, 41, 54, 49, 66, 64, 60, 52, 65, 44, 54, 58, 64, 59,  
48, 46, 56, 54, 51, 44, 52, 37, 48, 48, 57, 55, 36, 66, 44, 58, 70, 49, 55, 54, 54, 41, 4  
3, 47, 48, 46, 47, 47, 54, 60, 63, 62, 47, 62, 42, 42, 65, 58, 63, 40, 38, 67, 55, 52, 53,  
48, 47, 42, 61, 29, 41, 47, 32, 60, 62, 35, 60, 29, 44, 42, 60, 45, 56, 32, 66, 59, 40, 5  
7, 38, 44, 69, 43, 25, 46, 47, 47, 51, 44, 56, 38, 54, 51, 51, 55, 45, 54, 53, 40, 51, 49,  
47, 30, 46, 46, 53, 55, 44, 54, 40, 36, 49, 42, 80, 56, 42, 64, 50, 44, 71, 40, 48, 43, 4  
0, 41, 36, 60, 52, 53, 40, 46, 43, 43, 57, 46, 64, 40, 48, 64, 37, 33, 45, 50, 48, 55, 34,  
54, 60, 39, 45, 65, 48, 37, 41, 62, 43, 52, 43, 42, 63, 56, 42, 56, 36, 61, 55, 72, 50, 4  
3, 46, 53, 45, 66, 40, 64, 43, 35, 31, 59, 46, 63, 40, 60, 53, 46, 40, 60, 61, 44, 59, 54,  
56, 69, 48, 55, 52, 50, 46, 32, 60, 58, 50, 61, 59, 42, 49, 40, 59, 55, 51, 35, 45, 57, 5  
2, 35, 60, 37, 41, 53, 48]

E = [74, 22, 54, 104, 88, 73, 55, 67, 121, 48, 31, 73, 34, 54, 85, 96, 34, 43, 64, 65, 6  
3, 88, 82, 48, 86, 61, 74, 87, 89, 46, 45, 38, 113, 65, 72, 55, 61, 77, 67, 75, 101, 77,  
86, 70, 60, 40, 69, 44, 41, 25, 103, 87, 60, 96, 84, 54, 91, 47, 88, 97, 83, 55, 43, 53,  
104, 89, 45, 61, 73, 56, 62, 107, 79, 99, 69, 80, 85, 100, 51, 77, 78, 54, 46, 79, 76, 3  
2, 84, 106, 43, 70, 130, 85, 39, 87, 61, 61, 66, 58, 79, 51, 65, 85, 61, 87, 86, 63, 92,  
61, 65, 56, 65, 77, 89, 79, 77, 67, 31, 66, 71, 38, 96, 114, 5, 69, 81, 85, 52, 94, 41, 5  
1, 61, 58, 89, 56, 77, 84, 88, 76, 29, 57, 83, 48, 41, 65, 86, 32, 69, 58, 64, 48, 79, 85,  
64, 63, 63, 69, 85, 93, 127, 54, 64, 100, 71, 75, 68, 46, 76, 48, 54, 59, 62, 83, 76, 61  
, 61, 110, 48, 78, 75, 86, 61, 92, 66, 72, 65, 82, 54, 69, 59, 44, 89, 42, 76, 61, 92, 10  
1, 41, 66, 60, 80, 80, 68, 66, 60, 101, 48, 53, 84, 70, 76, 54, 82, 41, 46, 86, 100, 112,  
59, 94, 80, 88, 86, 68, 82, 77, 56, 47, 59, 74, 73, 90, 60, 47, 54, 39, 35, 99, 56, 113,  
37, 57, 82, 87, 49, 91, 84, 76, 53, 57, 39, 79, 72, 32, 86, 22, 45, 67, 79, 77, 61, 113,  
76, 70, 53, 65, 76, 42, 82, 57, 86, 52, 98, 76, 95, 31, 59, 79, 40, 101, 64, 53, 90, 63,  
70, 61, 98, 88, 60, 85, 80, 49, 55, 107, 64, 66, 49, 69, 88, 70, 33, 63, 94, 76, 67, 65,  
77, 56, 67, 67, 31, 62, 76, 58, 54, 65, 74, 38, 50, 59, 71, 45, 73, 83, 92, 96, 65, 85, 5  
8, 51, 37, 39, 47, 84, 65, 80, 127, 81, 49, 100, 78, 103, 54, 90, 91, 91, 45, 46, 83, 69,  
34, 24, 50, 37, 65, 59, 77, 82, 68, 58, 43, 82, 121, 120, 36, 73]

### Дополнительный вопрос на размышление:

Каковы общепринятые нормы посещаемости развлекательных центров, исходя из которых их можно считать рентабельными?

### Решение:

```
import matplotlib.pyplot as plt
plt.bar(['A', 'B', 'C', 'D', 'E'], [sum(A), sum(B), sum(C), sum(D), sum(E)])
```

```
plt.title('Ответ D');
```

### Решение без построения графика:

Вывести сумму по всем массивам и посмотреть минимальную посещаемость.

```
print(sum(A), sum(B), sum(C), sum(D), sum(E))
```

**Ответ:** центр отдыха с минимальной посещаемостью под буквой D (рис.3).

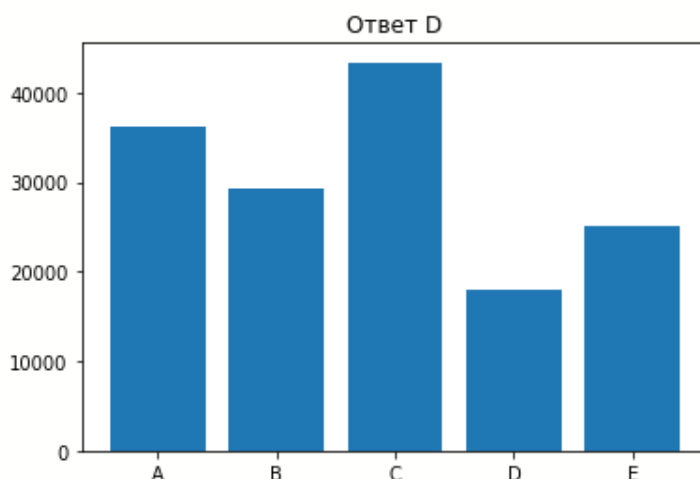


Рис. 3. Посещаемость центров отдыха

### Задача 3. Как найти проходное место в городе?

#### Условие.

Проходное место - место, где бывает много людей. Успех нового проекта городской инфраструктуры во многом зависит от того, как далеко от такой точки находится объект. Если мы найдем точку города, в которой больше всего людей, мы сможем дать ответы на следующие вопросы:

- Как далеко от этой точки находится центр?
- Почему люди из этой точки не доходят до центра?

Вам нужно проанализировать трехмерный массив со следующей структурой (рис. 4):

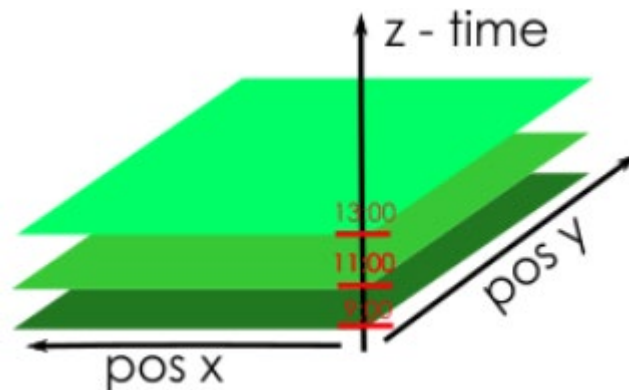


Рис. 4. Структура трехмерного массива

По оси Z - время в городе; X, Y - позиция в городе.

По этим координатам можно посмотреть количество людей в определенной точке в определенное время. Таким образом если мы усредним данные по времени и в полученном массиве найдем ячейку с максимальным значением — это точка города, где за сутки проходит больше всего людей - "проходное" место.

Ваша задача - найти эту точку. Ответ запишите в формате: x, y (без скобок). Для генерации данных используйте функцию, описанную ниже.

```
def generate():
    import numpy as np
    np.random.RandomState(42)
    r = []
    for i in range(24):
        x = np.linspace(-90, 10, 100)
        y = np.linspace(-50, 50, 100)
        x, y = np.meshgrid(x, y)
        f = 7*(-x**2)+1*(-y**2)+100*np.sin(i/48*2*np.pi)
        f[f<10]=np.random.random(f[f<10].shape)*40 +30
        r.append(f)
    return [[[round(el, 3) for el in r]for r in m] for m in r]
```

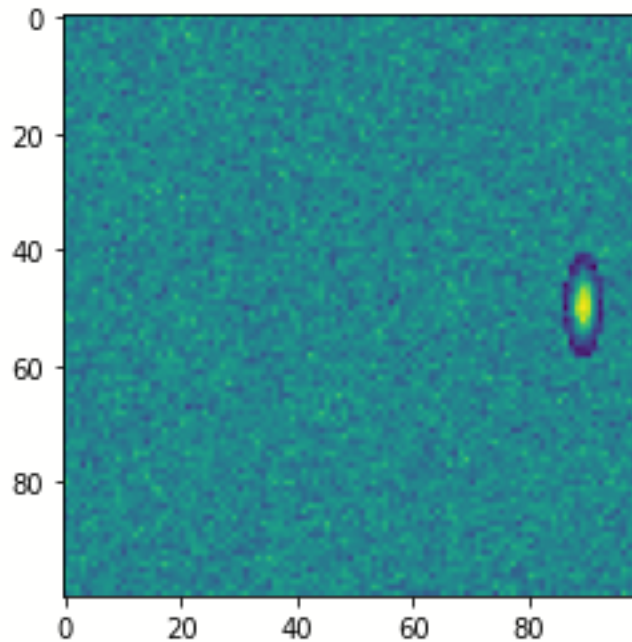
## Решение

#Предполагается решение в интерактивной тетрадке (рис. 5)

```
import matplotlib.pyplot as plt
import numpy as np
```

```
img = np.mean(generate(), axis = 0)
plt.imshow(img)
np.where(img == np.max(img))
```

**Ответ:**(array([50]), array([89]))



*Рис.5. Графическое представление решения задачи 3*

**Ответ** 89, 50 или 89, 49

#### **Задача 4. Маршруты**

##### **Условие**

На картинке (рис. 6, 7) изображена схема маршрутов и длина путей между пунктами.

Известно, что точка Е — это область, где живет большинство людей. Ж - исследуемый торговый центр. Определите количество маршрутов из района с максимальным населением до развлекательного центра, укажите длину наименьшего из найденных маршрутов.

В ответе запишите два целых числа через пробел. Первое - количество маршрутов, позволяющих попасть из точки Е в точку Ж, второе – длину минимального маршрута в километрах.

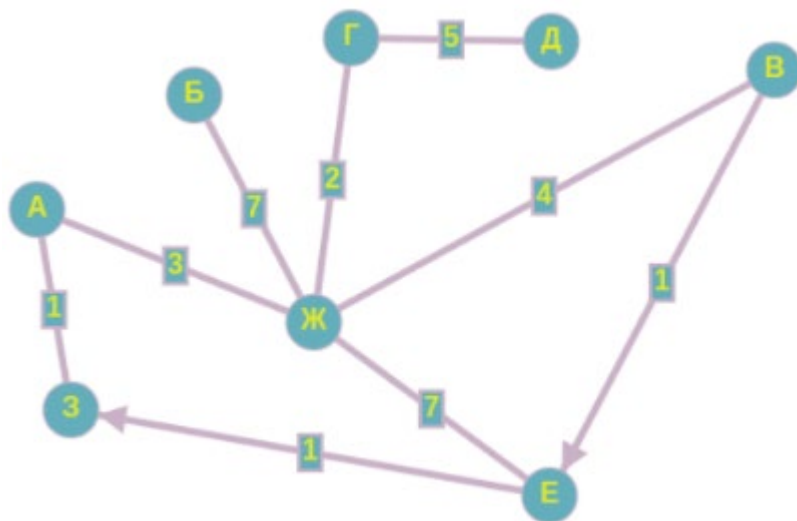


Рис. 6. Граф маршрутов между ключевыми точками

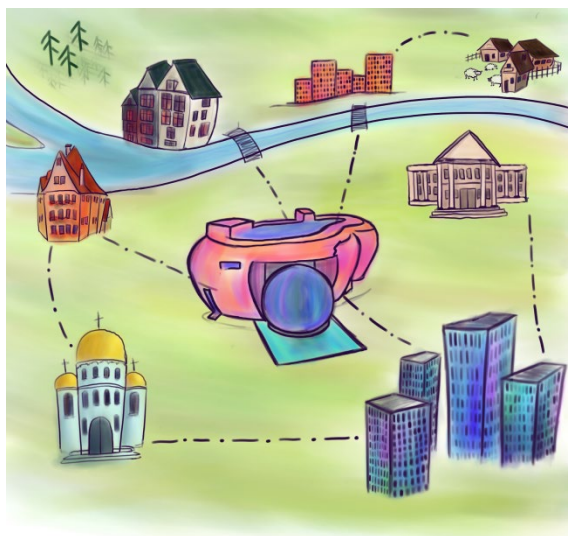


Рис. 7. Схема расположения ключевых точек города

Обратите внимание, что стрелка указывает направления движения: там, где она есть - дорога односторонняя, где стрелок нет - движение имеет

двунаправленный характер. Цифра на ребрах графа(на прямой, соединяющей маршрутные пункты) обозначает расстояние между ними в километрах.

**Дополнительный вопрос на размышление:**

Относительно площади города, можно ли сказать, что даже минимальный маршрут слишком длинный?

**Решение.**

Длину маршрутов, согласно графу, выпишем в таблицу:

Маршрут 1	Маршрут 2	Маршрут 3
$1+1+3=5$	7	нет

Из точки Е существует два варианта движения к точке Ж. Третий вариант нужно исключить, так как из точки Е в точку В противоположное движение.

**Ответ:** 2 маршрута, 5 км минимальное расстояние

### Задача 5. Анализ сообщений в социальных сетях

**Условие.**

Теперь мы знаем, что маршрутов ведущих в развлекательный центр мало. Тем не менее, это не может быть главной причиной, из-за чего люди туда не ходят.

Давайте посмотрим сообщения в социальных сетях, оставленные людьми с улиц, расположенных по соседству с новым развлекательным центром улиц. Возможно, настроение улиц сказывается на низкой посещаемости места отдыха?

Постройте облако слов на основе обобщенного текста, составленного из таких сообщений. В ответ запишите 3 слова, которые занимают самое большое место в облаке. При построении облака слов уберите следующие слова ["на", "все", "эта", "не", "вчера", "уже"].



Сообщения (сохранены стилевые характеристики текстов социальных сетей):



Рис. 8. Иллюстрация к задаче 5

Ох уж эта стройка! Мало того, что постоянно дождь, так еще и эта грязь на улице. Никуда не годится.

Ехал вчера на машине мимо. Думал не проеду. Могли бы уже завершить эти строительные работы. Ведут стройку уже больше года.

Был вчера в новом развлекательном центре. Мне все понравилось, всем советую. Ноги правда все испачкал пока дошел, но все равно круто.

Дожди-дожди-дожди((((((( Устала ходить на работу, особенно в такую погоду.

Кто-то стащил со стройки цемент, ахаха)).

Строится новый жилой комплекс Стрела! Спишите купить квартиру за смешную цену. Стрела приглашает на круглый стол «Социальные объекты для новых районов: как синхронизировать стройку?

Ура солнце! Лол впервые за 100000 лет.

**Решение**

```
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
from collections import Counter
import matplotlib.pyplot as plt
```

```
comments = """"
```

```
Ох уж эта стройка! Мало того, что постоянно дождь, так еще и эта грязь на
улице. Никуда не годится. Ехал вчера на машине мимо. Думал не проеду. Могли
бы уже завершить эти строительные работы. Ведут стройку уже больше года.
Был вчера в новом развлекательном центре. Мне все понравилось, всем
советую. Ноги правда все испачкал пока дошел, но все равно круто. Дождди-
дожди-дожди(((( Устала ходить на работу, особенно в такую погоду. Кто-то
сташил со стройки цемент, ахаха)). Строится новый жилой комплекс Стрела!
Спишите купить квартиру за смешную цену. Стрела приглашает на круглый
стол «Социальные объекты для новых районов: как синхронизировать стройку?
Ура солнце! Лол впервые за 100000 лет.""""
```

```
#Выделяем слова, которые точно не должны использоваться при построении о
блака слов!
```

```
stopwords = set(STOPWORDS)
stopwords.update(["на", "все", "эта", "не", "вчера", "уже"])
```

```
#Создаем облако
```

```
wordcloud = WordCloud(stopwords = stopwords, background_color="white").gener
ate(comments)
```

```
# Рисуем результат
```

```
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```

Получившееся облако слов показано на рисунке 9.

**Ответ:** СТРОЙКУ, ДОЖДИ, СТРЕЛА

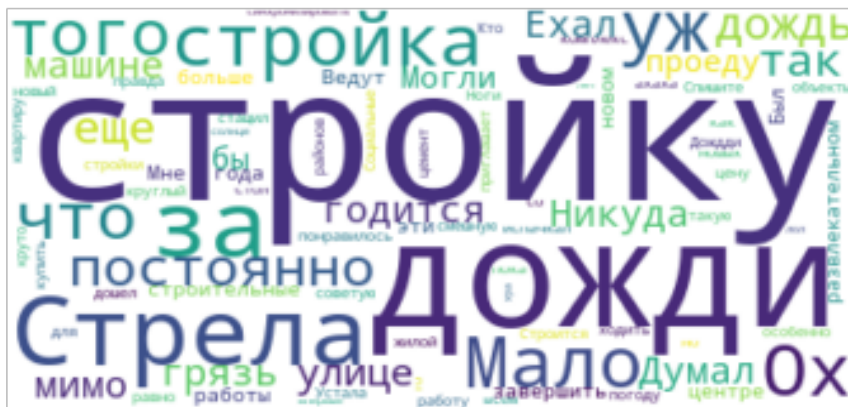


Рис. 9. Облако слов

## Задача 6. Погода в городе

### Условие

Третий и четвертый пункт дают повод задуматься. Стройка рядом, это не так страшно...не так, когда погода в городе солнечная, а дороги сухие без грязной жижи.

Постройте диаграмму-пирог, на которой можно посмотреть, соотношение ненастных дней и дней с хорошей погодой.

Для это нам понадобится два массива: в первом массиве  $t$  - указана температура воздуха за день, во втором  $w$  - количество осадков в этот же день.

Погода определяется по условию:

**Хорошая** если:  $t > 20, w = 0$

**Прогулочная:**  $t > 20, w \neq 0, w < 2$

**Пасмурно:**  $t < 20 \text{ and } t > 15, w \neq 0 \text{ and } w < 2$

**Плохая:**  $t < 15 \text{ and } t > 7 \text{ and } w > 2$

Определите по получившейся диаграмме погодные условия, преобладающие на территории города. Какова длительность в днях каждого погодного периода?

### Входные данные:

t = [19.0, 15.1, 27.44, 23.57, 18.67, 15.23, 17.72, 17.46, 8.33, 19.59, 15.41, 16.09, 2  
7.07, 21.46, 10.73, 9.26, 27.61, 19.79, 27.0, 22.29, 9.91, 10.02, 16.34, 18.54, 15.33,  
11.79, 23.92, 25.92, 17.58, 15.24, 8.68, 10.4, 15.48, 23.72, 19.71, 19.24, 28.8, 21.72  
, 11.66, 19.88, 27.6, 17.76, 26.95, 18.95, 18.45, 24.75, 11.47, 15.79, 19.38, 15.8, 18.  
31, 18.28, 20.44, 15.79, 23.76, 23.08, 15.65, 21.36, 19.59, 27.04, 8.11, 9.88, 18.94,  
20.9, 29.8, 28.03, 9.36, 27.68, 9.67, 17.19, 18.28, 20.38, 27.58, 21.6, 25.13, 15.81, 2  
2.92, 18.54, 8.19, 9.49, 18.25, 29.46, 17.55, 9.31, 18.19, 15.45, 23.36, 19.99, 29.77,  
20.61, 10.64, 29.48, 20.99, 17.73, 16.92, 17.43, 26.69, 19.66, 19.57, 28.44, 25.18, 2  
1.61, 17.14, 16.81, 7.96, 17.85, 27.69, 18.88, 15.29, 17.23, 18.34, 19.7, 19.74, 19.5,  
16.64, 23.31, 19.88, 11.1, 19.27, 21.49, 11.68, 16.33, 26.74, 11.5, 8.96, 18.04, 24.91  
, 7.32, 16.25, 19.37, 8.82, 27.72, 7.77, 19.55, 28.42, 16.66, 18.6, 22.68, 27.95, 15.54  
, 18.7, 19.12, 7.33, 26.12, 16.51, 18.17, 19.18, 19.96, 25.35, 18.59, 8.18, 16.37, 7.22  
, 17.47, 18.43, 8.12, 17.92, 22.16, 25.49, 17.73, 19.25, 9.51, 7.96, 23.99, 24.17, 15.4  
, 22.77, 19.9, 17.49, 19.55, 21.26, 23.82, 7.34, 24.64, 18.92, 16.39, 26.34, 8.93, 19.5  
8, 19.42, 8.35, 8.77, 28.77, 17.94, 16.48, 7.1, 28.42, 19.45, 7.9, 15.47, 16.95, 15.9, 1  
5.66, 17.04, 15.08, 19.73, 28.99, 8.79, 23.42, 18.69, 28.52, 16.67, 11.99, 16.14, 25.4  
4, 16.7, 21.05, 9.77, 16.13, 16.09, 24.03, 16.15, 17.07, 11.66, 19.23, 25.76, 19.16, 1  
7.22, 15.2, 26.48, 23.01, 16.4, 23.91, 22.34, 24.86, 16.52, 23.57, 19.28, 8.29, 16.57,  
16.09, 17.3, 17.81, 26.82, 24.59, 16.85, 17.02, 11.53, 27.99, 15.35, 15.97, 20.04, 15.  
16, 9.96, 11.03, 16.93, 19.19, 7.35, 23.38, 28.54, 27.33, 9.49, 9.04, 8.82, 7.46, 16.29  
, 16.29, 25.94, 23.67, 23.99, 16.11, 15.05, 25.62, 19.78, 15.63, 20.74, 24.02, 17.46,  
28.7, 17.2, 8.09, 16.46, 8.07, 16.07, 11.47, 20.62, 25.89, 21.57, 15.72, 10.64, 18.82,  
17.57, 18.64, 16.11, 23.97, 16.61, 18.03, 18.11, 28.35, 16.08, 16.11, 27.29, 15.92, 1  
5.93, 25.63, 15.67, 19.61, 7.27, 24.58, 11.39, 9.08, 16.66, 27.38, 22.44, 18.52, 24.23  
, 19.1, 8.17, 17.29, 15.65, 25.89, 29.7, 16.79, 19.86, 21.01, 9.73, 19.21, 7.04, 19.26,  
10.3, 23.39, 10.65, 19.79, 17.08, 15.85, 7.12, 19.97, 17.3, 15.0, 18.47, 17.85, 19.57,  
11.13, 11.32, 19.73, 22.29, 22.93, 18.25, 16.51, 18.81, 16.26, 28.87, 25.38, 17.96, 9.

62, 10.52, 19.61, 15.24, 26.87, 10.75, 19.71, 18.01, 8.43, 16.05, 11.29, 8.36, 18.56, 11.38, 16.67, 25.65, 19.67, 26.85, 15.46, 15.92, 19.18]

w = [1.299, 0.408, 0.0, 0.386, 1.252, 0.599, 1.198, 0.15, 2.458, 1.529, 0.165, 0.951, 0.829, 0.0, 2.012, 2.325, 0.996, 1.676, 1.913, 1.584, 2.33, 2.75, 1.855, 1.473, 1.762, 2.933, 1.256, 0.0, 1.904, 1.185, 2.476, 2.069, 0.321, 0.0, 1.836, 0.89, 1.147, 0.934, 2.989, 0.262, 0.0, 0.587, 1.108, 1.537, 1.568, 1.884, 2.463, 1.41, 1.425, 0.95, 1.851, 1.16, 0.106, 1.443, 0.932, 0.088, 1.478, 0.648, 1.683, 0.0, 2.328, 2.964, 1.132, 0.567, 0.075, 0.839, 2.517, 0.0, 2.648, 1.382, 0.111, 1.284, 1.547, 0.322, 1.533, 0.854, 0.0, 1.435, 2.438, 2.837, 1.859, 0.833, 0.633, 2.495, 1.655, 0.714, 0.562, 0.972, 0.364, 1.904, 2.174, 1.254, 1.457, 1.673, 0.702, 0.646, 0.0, 0.779, 0.151, 0.0, 0.0, 1.458, 1.54, 1.643, 2.122, 0.364, 0.0, 1.072, 1.146, 1.251, 0.064, 1.464, 1.962, 0.253, 1.276, 1.573, 1.13, 2.57, 0.236, 0.0, 2.682, 0.181, 0.088, 2.529, 2.348, 0.955, 1.101, 2.274, 1.151, 1.566, 2.177, 0.733, 2.21, 0.67, 0.285, 0.516, 0.145, 0.0, 0.0, 0.637, 1.616, 1.494, 2.748, 0.294, 1.366, 0.058, 1.86, 1.421, 1.442, 1.804, 2.561, 1.524, 2.494, 1.429, 1.16, 2.68, 0.286, 0.042, 0.184, 1.611, 0.655, 2.076, 2.683, 0.489, 0.0, 1.81, 0.0, 1.257, 1.961, 1.024, 0.0, 1.067, 2.625, 0.0, 1.014, 0.763, 0.0, 2.248, 1.173, 1.097, 2.416, 2.159, 0.0, 0.709, 1.864, 2.051, 1.662, 0.452, 2.361, 1.443, 0.91, 0.503, 0.743, 1.048, 1.62, 1.655, 1.594, 2.484, 0.437, 0.393, 1.656, 0.256, 2.321, 1.197, 1.585, 0.614, 0.555, 2.128, 0.109, 1.563, 1.065, 0.357, 0.859, 2.981, 1.848, 0.0, 1.574, 1.161, 1.309, 0.62, 1.234, 0.495, 0.009, 1.311, 1.091, 1.586, 0.368, 0.6, 2.354, 1.799, 0.056, 0.182, 0.458, 0.0, 1.65, 0.722, 0.844, 2.316, 0.0, 0.272, 0.527, 0.0, 0.908, 2.666, 2.954, 0.105, 1.797, 2.187, 0.399, 1.047, 0.823, 2.217, 2.928, 2.473, 2.986, 1.034, 0.345, 0.0, 0.488, 0.0, 1.162, 1.509, 0.045, 0.691, 1.018, 1.175, 0.37, 0.626, 0.157, 1.749, 2.828, 0.898, 2.974, 0.371, 2.977, 0.0, 0.0, 0.0, 0.057, 2.35, 1.68, 0.604, 0.023, 0.161, 0.445, 1.689, 0.676, 0.781, 0.0, 0.658, 0.715, 0.0, 1.971, 0.871, 0.0, 1.57, 0.769, 2.918, 0.121, 2.834, 2.346, 1.81, 0.098, 1.595, 1.951, 0.976, 1.164, 2.055, 1.109, 0.576, 0.

0, 0.852, 0.249, 1.517, 0.0, 2.034, 0.79, 2.645, 0.981, 2.015, 1.505, 2.655, 0.403, 0.59, 0.561, 2.753, 0.755, 0.454, 0.381, 1.231, 1.311, 0.457, 2.286, 2.943, 1.843, 1.38, 1.88, 1.66, 0.114, 0.571, 0.915, 0.0, 0.854, 1.719, 2.676, 2.511, 1.399, 1.137, 0.251, 2.507, 1.693, 1.301, 2.329, 1.192, 2.34, 2.818, 0.037, 2.415, 0.926, 1.058, 0.583, 0.0, 0.251, 1.414, 0.305]

## Решение

Предполагается решение в интерактивной тетрадке по примеру из задач самоподготовки

```
weather_year = [0,0,0,0]
i=0
while(i<len(t)):
if t[i] >20and w[i] == 0:
weather_year[0]+=1
elif t[i]>20and w[i]!=0and w[i]<2:
weather_year[1]+=1
elif (t[i] <20and t[i] >15) and (w[i] !=0and w[i] <2):
weather_year[2]+=1
elif t[i] <15and t[i]>7and w[i]>2:
weather_year[3]+=1
i+=1
explode = (0, 0, 0.1, 0)
doies = ["хорошая {0}".format(weather_year[0]), "прогулочная
{0}".format(weather_year[1]), "пасмурная {0}".format(weather_year[2]), "плохая
{0}".format(weather_year[3])]

importmatplotlib.pyplotasplt
plt.figure(figsize = (10,10))
plt.pie(weather_year, explode=explode, labels=doies, shadow=True, startangle=90)
plt.show()
```

Полученная диаграмма представлена на рисунке 10.

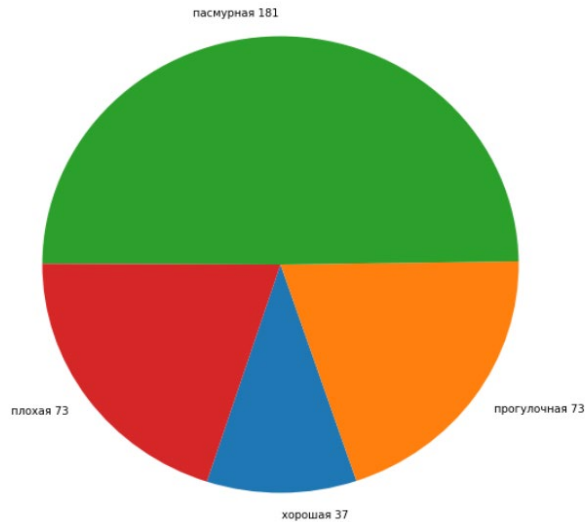


Рис. 10. Диаграмма погоды

**Решение без отрисовки диаграммы:**

```

weather_year = [0,0,0,0]
i=0
while(i<len(t)):
if t[i] >20and w[i] == 0:
weather_year[0]+=1
elif t[i]>20and w[i]!=0and w[i]<2:
weather_year[1]+=1
elif (t[i] <20and t[i] >15) and (w[i] !=0and w[i] <2):
weather_year[2]+=1
elif t[i] <15and t[i]>7and w[i]>2:
weather_year[3]+=1
i+=1
print(weather_year)

```

**Ответ:** преобладающая погода пасмурная.

Количество дней и тип погоды

Пасмурная 182  
Плохая 73  
Прогулочная 73  
Хорошая 37

### Задача 7. Выбираем место под новый жилой комплекс

#### Условие

Новый развлекательный центр в городе — это хорошо. Но надо понимать, что вблизи развлекательного комплекса должны находиться густонаселенные жилые районы с доступными транспортными развязками.

Если бы мы могли построить еще несколько жилищных комплексов, расположенных вблизи маршрутов, ведущих к развлекательному центру, то удалось бы увеличить и поток его посетителей.

По карте, которая приведена ниже (рис. 11), найдите территориальный пункт, от которого идет больше всего маршрутов к нашему развлекательному центру. В ответе укажите букву такого пункта.



Рис. 11. Карта города

#### Решение

Нарисуем на основе карты и информации из предыдущих заданий граф маршрутов, по которым люди могут перемещаться между ключевыми точками города (рис. 12).



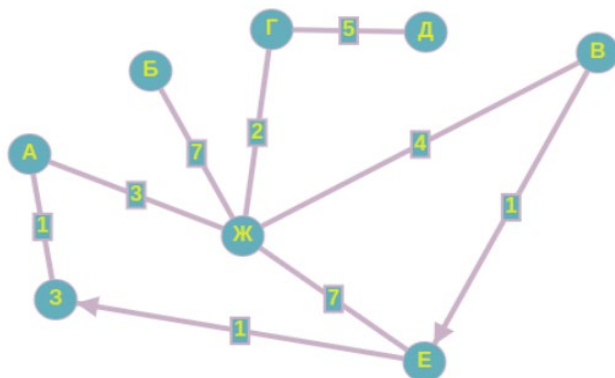


Рис. 12. Граф маршрутов

Развлекательный комплекс находится в точке Ж.

Таблица 2. Количество маршрутов к точке Ж

Из точки	Количество маршрутов к Ж
А	1
Б	1
В	3
Г	1
Д	1
Е	2
З	1

**Ответ:** Наибольшее количество маршрутов идет из пункта В

### Задача 8. Время работы развлекательного центра

#### Условие

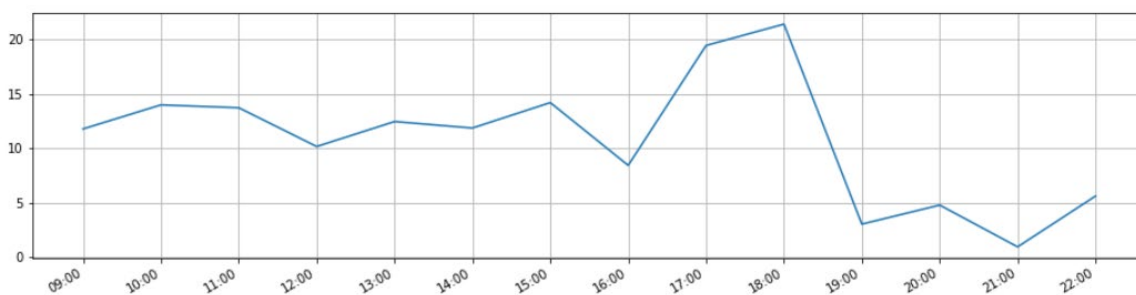
Так-с! Начало делу положено. Стоит отметить, что не менее важно определиться с временем работы центра отдыха. Надо понимать - время деньги. Если неправильно выбрать время его работы, то народу будет приходить мало, а денег на поддержание инфраструктуры будет тратиться много. С учетом, что на обслуживание центра планируется тратить в день 400 тысяч условных денежных единиц, а в час – стоимость работы по обслуживанию составляет - 50 тысяч условных денежных единиц, стажерам компании AltFuture следует

выяснить по графикам местонахождения и времяпрепровождения горожан, оптимальные часы работы центра.

**Ниже дано три графика:**

Количество людей, пребывающих с 8.00 до 22.00 на улицах, которые расположены в непосредственной территориальной близости от пункта Ж

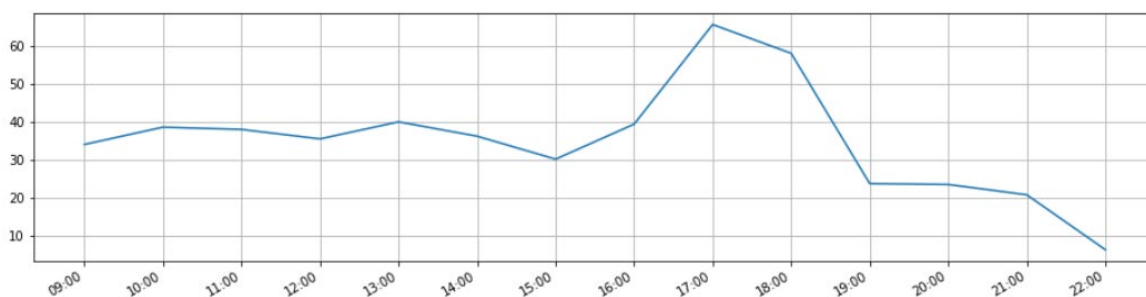
1



*Рис. 13. Количество людей в непосредственной близости от пункта Ж*

Количество людей, находящихся непосредственно в точке расположения развлекательного комплекса с 8.00 до 22.00

2



*Рис. 14. Количество людей в точке расположения комплекса*

Количество людей, использующих с 8.00 до 22.00 маршруты следования по направлению к развлекательному комплексу, которые отправляются из района нового жилищного комплекса (см. задание 7)

3

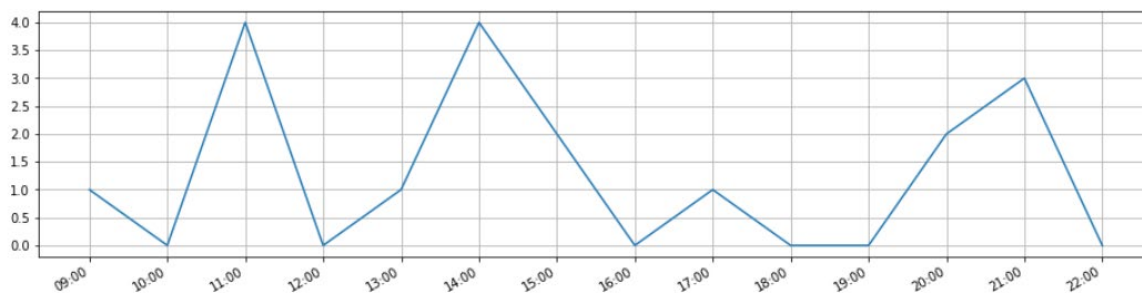


Рис. 15. Количество людей, следующих по маршруту в направлении пункта Ж

Проанализируйте графики. Определите оптимальный временной интервал работы центра, исходя из числа потенциальных посетителей и расходов на обслуживание центра отдыха.

Напишите ответ слитным четырехзначным числом. Например, с 9 утра до 21 вечера - 0921.

### Решение

По условию задачи у комплекса в планах тратить не более 400 000 условных денежных единиц. И так как час работы центра составляет 50 000 денежных единиц, это означает, что центр отдыха будет работать 8 часов в день.

Теперь посмотрим на графики:

- Исходя из первого графика, мы можем заметить, что в 18:00 идет резкое уменьшение численности людей в точках близких к исследуемому центру отдыха и после этого времени держится на довольно низкой отметке ~5 человек, при таких количествах посетителей, вряд ли комплекс будет окупаться.
- Но втором графике, видно, что людей непосредственно возле центра отдыха больше пяти, но после 19:00 меньше, чем за весь день. Максимальное количество людей приходится на 17:00 — значит в это время комплекс точно должен работать.

- На третьем графике видно, что пики активности движения от нового жилого комплекса к центру отдыха приходятся на 11:00 и 14:00. Захватим и это время для работы центра.

После этих наблюдений интервал работы комплекса определили как с 11:00 до 17:00. Это только 6 часов работы, поэтому рассмотрим численность людей:

- в 09:00 в 10:00 для определения начала рабочего дня
- в 18:00 и 19:00 для определения окончания рабочего дня

На втором графике видно, что больше людей в 10:00 и 18:00, но нужно учитывать еще и тех, кто близко. Считаем примерное количество людей в выбранные промежутки времени:

$$9 \text{ утра: } 12 + 35 + 10 = 57$$

$$10 \text{ утра: } 14 + 40 + 0 = 54$$

$$18 \text{ вечера: } 22 + 60 + 0 = 82$$

$$19 \text{ вечера: } 4 + 24 + 0 = 28$$

По результатам подсчетов видно, что если выбрать время работы центра с 10:00 до 18:00, то потенциальных посетителей будет больше, чем в другие промежутки времени.

**Ответ:** оптимальный режим работы центра с 10 утра до 18 вечера

### Задача 9. Сколько "счастья" надо?

#### Условие

Мы выбрали время работы центра. А как насчет комфорта, сервиса, развлечений?

Давайте, определим количество возможных предприятий фудкорта, развлечений и торговли.

Как вы уже знаете, у нашего развлекательного комплекса есть ограничение по тому, сколько он может затратить на обеспечение своей инфраструктуры (см. задачу 8). Кроме того, мэр города и городской совет предполагали, что новый объект станет не просто развлекательным комплексом, а будет центром "притяжения" горожан и гостей - надо, чтобы в нем посетителей было больше, чем в других центрах. Это значит, объектов, позволяющих людям чувствовать себя счастливыми, в этом центре должно быть достаточно много. Осталось узнать, достаточно много — это сколько?

Будем считать, что на обслуживание фудкортов за 8 часов тратится 10 000 условных денежных единиц (один фудкорт), на аттракционы за 8 часов расходуется 60 000 условных денежных единиц (одна площадка), торговые павильоны отнимает за 8 часов 2 000 условных денежных единиц (один павильон).

Также социологи считают, что фудкорты прибавляют 1000 очков "счастья". Аттракционы +3000 очков "счастья". Торговые павильоны +600 очков "счастья". Сколько необходимо фудкорт, площадок с аттракционами и торговых павильонов, чтобы уложиться в выделенную на обслуживание инфраструктуры денежную сумму, и обойти по очкам "счастья" другие развлекательные комплексы. Усредненное количество очков счастья по другим центрам: 20020, 29000, 35000, 23000.

Для подбора оптимальных вариантов рекомендуется создать интерфейс для манипулирования основными параметрами: food, market, joy (см. материалы для самоподготовки, задание 5). Двигайте ползунки и меняйте параметры, чтобы получить оптимальный вариант: расходы-"счастье".

В ответе запишите три числа через пробел: соответствующие значения для food, market и joy.

**Решение**

Ответ может быть не самым оптимальным, но должен соответствовать следующим условиям:

- Общий бюджет не должен превышать 400 тысяч условных денежных единиц.
- Нужно заработать больше 35000 очков "счастья".

При следующих значениях предприятий:

- На фудкорты тратится 10 000 условных денежных единиц и +1000 очков "счастья".
- Торговые павильоны отнимает за 8 часов 2 000 условных денежных единиц и +600 очков "счастья".
- На аттракционы расходуется 60 000 условных денежных единиц и +3000 очков "счастья".

Ответом, например, может служить перечень:

- 10 фудкортов (100 000 денежных единиц и 10000 "счастья")
- 28 торговых предприятий (56 000 денежных единиц и 16 800)
- 4 аттракциона (240 000 и 12000)

Итого: при затратах условных денежных единиц 396 000 получаем 38 800 очков "счастья", что больше, чем очки счастья в других центрах. Цель выполнена!

Осталось еще 4 000 денежных единиц и можно добавить два торговых павильона, тогда мы потратим весь бюджет центра и получим 40 000 очков "счастья". Данный ответ (10 фудкортов, 30 торговых павильонов и 4 аттракциона) тоже будет считаться верным.

**Решение интерактивной тетрадке.**

```
def counts_of_points(foods = 10, market = 10, joy = 10):
```

```
    mfoods = foods*10000
    mmarket = market*2000
    mjoy = joy*60000
```

```

spends = [mfoods, mmarket, mjoy]
maxspend = 400000
title_1 = "Текущие траты %d!" % sum(spends) if sum(spends) < maxspend else "Т
екущие траты %d! Превышают лимит 400-000" % sum(spends)

hfoods = foods*1000
hjoy = joy*3000
hmarket = market*600
max_happy = hfoods+hjoy+hmarket

other_centers = [20020, 29000, 35000, 23000, max_happy ]
explode = explode = (0, 0, 0, 0, 0.1)
title_2 = "Счастья слишком мало!" if max(other_centers[:-1]) > other_centers [-1]
else "Счастья сколько надо!"

plt.figure(figsize = (10, 4))
plt.subplot(1,2,1)
plt.pie(spends,labels=spends, autopct='%1.1f%%',
        shadow=True, startangle=90)
plt.title(title_1)

plt.subplot(1,2,2)

plt.pie(other_centers,labels=other_centers, explode = explode, autopct='%1.1f%%',
shadow=True, startangle=90)
plt.title(title_2)
plt.tight_layout()
plt.show()
import matplotlib.pyplot as plt
from ipywidgets import interact, interactive, fixed, interact_manual
from IPython.display import display

#counts_of_points()
w = interactive(counts_of_points, foods = 10, market = 28, joy = 4)

display(w)

```

Функция проверки для [stepik.org](http://stepik.org), если для организации автоматической проверки программ использована эта платформа.

```
def check(answer):
    foods, market, joy = list(map(int,answer.split()))

    mfoods = foods*10000
    mmarket = market*2000
    mjoy = joy*60000
    spends = [mfoods, mmarket, mjoy]
    maxspend = 400000

    hfoods = foods*1000
    hjoy = joy*3000
    hmarket = market*600
    max_happy = hfoods+hjoy+hmarket

    other_centers = [20020, 29000, 35000, 23000, max_happy ]
    explode = explode = (0, 0, 0, 0, 0.1)
    title_2 = "Счастья слишком мало!" if max(other_centers[:-1]) > other_centers[-1] else "Счастья сколько надо!"

    if sum(spends)>maxspend:
        return False

    if max(other_centers[:-1]) > other_centers[-1]:
        return False

    return True
```



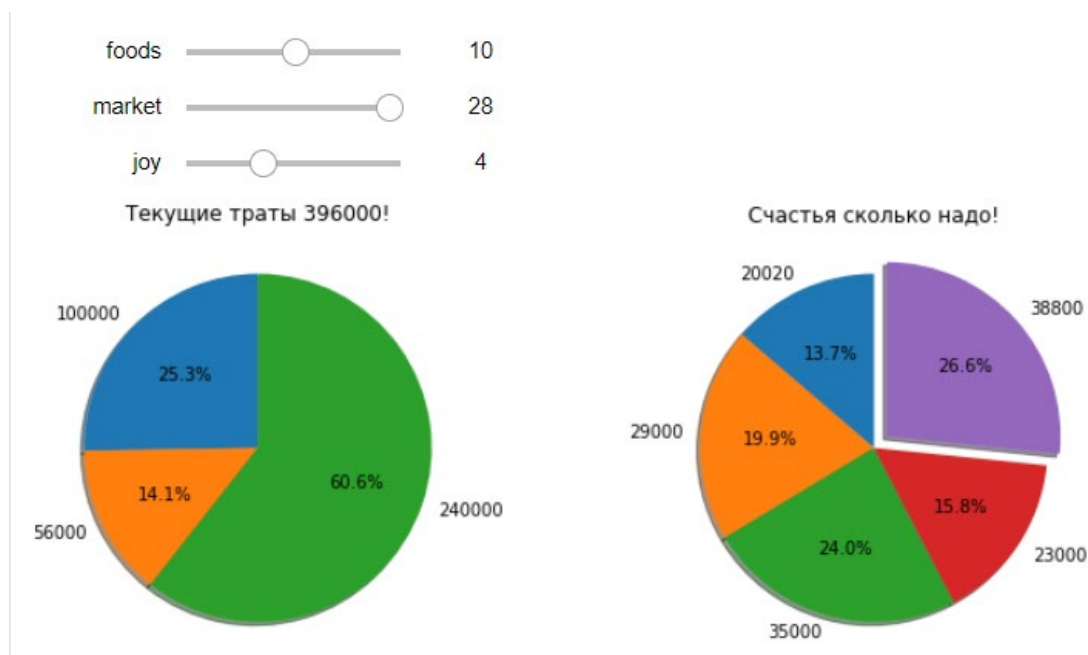


Рис. 16. Результат выполнения программного кода.

## Задача 10. Конкурс заявок

### Условие

Стажеры определили оптимальное количество предприятий, которые стоит разместить в новом развлекательном комплексе. Теперь у мэра возникла новая проблема.

Список заявок на открытие своих представительств в центре отдыха больше, чем установленное в задаче 9 рекомендуемое количество. Надо решить, чьи заявки лишние.

На графике представлены интересы жителей города. Пользуясь графиком и опираясь на мнение горожан, постройте список предприятий, заявки от которых не будут реализованы.

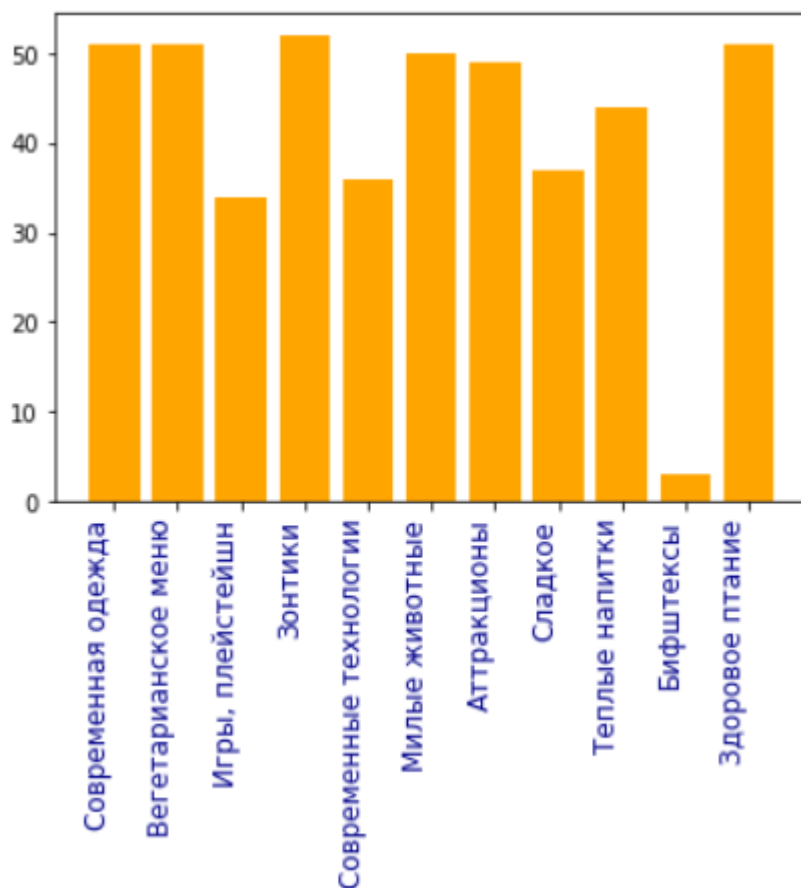


Рис. 17. График интересов жителей города.

Исходя из графика, нужно определить какие компании не пройдут отбор и соответственно не представят свои предприятия в новом развлекательном комплексе:

- Кото-Кафе Ламповое. Можете согреться в пасмурную, выпить горячего шоколада, погладить котиков.
- VR зона. Атракционы смешанной, дополненной и виртуальной реальности!
- Магазин зонтов Оле Лукое. Самый широкий ассортимент в городе.
- Облачный парк - парк аттракционов прямо на крыше города. 245 видов качелей собранных из разных уголков вселенной. А также горки и бассейн.
- магазин одежды "Нежность". Все что нужно вашему ребенку.

- Ресторан "Две Шпаги". Мясо и мясо и еще много мяса.
- Магазин овощей и фруктов.
- Магазин сына подруги твоей мамы - одежда для стильных тинейджеров.

## **Решение**

Заявки, которые необходимо отклонить:

- Облачный парк - парк аттракционов прямо на крыше города. 245 видов качелей собранных из разных уголков вселенной. А также горки и бассейн
- Ресторан "Две Шпаги". Мясо и мясо и еще много мяса

Исходя из графика, можно сказать, что жителям не интересны Бифштексы, поэтому стоит отклонить заявку ресторана "Две Шпаги".

Заявку Облачного парка стоит тоже отклонить, хоть и интерес к аттракционам есть, но исходя из прошлого задания про погоду в городе, можно точно сказать, что аттракционы на крыше центра будут пустовать/закрывают большую часть года. К тому же, в противовес данной заявке есть VR зона, которая удовлетворит потребности горожан по аттракционам.

Остальные заявки попадают в интересы горожан.

## **Второй модуль индивидуального этапа.**

### **Постановка проблемы для участников.**

Современный город — это экосистема. Мэр и администрация г. Энск стараются создавать в нем наиболее благоприятные условия для жизни.

При реконструкции города решили уделить особое внимание парковым зонам и местам отдыха. Они являются сердцем города. Парковые зоны способствуют улучшению качества воздуха и являются средой обитания представителей флоры и фауны. Также парковые зоны способствуют сплочению городского населения и повышению качества его жизни. Перед

Junior-стажерами компании AltFuture поставлена задача исследования городского парка Энска "Роднички". Парк был построен еще в начале прошлого века и с тех пор никаких работ по его улучшению предпринято не было. Место отдыха "Роднички" расположено на территории прекрасного природного массива, но, к сожалению, по прошествии многих лет концепция парка устарела: его аллеи опустели, он больше не является сердцем города. Пришло время нового содержания для этого замечательного места.

Новый директор "Родничков" хочет сделать парк более молодежным. Добавить динамики! Пусть молодежь оторвется от электронных развлечений, будет больше времени проводить на свежем воздухе, общаться, знакомиться и задумываться о будущем своего города.

Нашим Junior-стажерам необходимо в предстоящем исследовании решить 10 задач. В частности, предлагается исследовать различные парки культуры и отдыха и исходя из этого, а также опираясь на предпочтения горожан, придумать как можно улучшить парк. Также не стоит забывать особенность г. Энска: как было выяснено в прошлый раз, это пасмурный город со значительным количеством осадков.

**Задача 1. Ищем аналоги**

**Условие.**

Один из вариантов модернизации парка — это поиск готовых решений среди парков других городов. Надо найти городские парковые зоны с наибольшей популярностью. Решение по оснащению, зонированию, техническому обеспечению парковой зоны может быть перенесено полностью из одного города в другой, но только в том случае, если город идентичен по ряду параметров Энску. И, в первую очередь, это касается погодных условий. Будем искать города аналог Энска.

*Вам дана таблица следующего формата:*

День	Город	Погода
1	1	1
1	2	3
2	3	4
2	2	1
3	4	2

Первый столбец — это день года.

Второй — код города.

Третий — оценка погодных условий в городе в определенный день.

Вам необходимо найти 3 города с наиболее схожими с Энскими погодными условиями. Средняя годовая оценка погодных условий в Энске составляет 2.12 балла.

Примечание. Для задачи таблица дается в двух форматах `pythonDict` и `csv`. Решить задачу можно как написав программу, так и с помощью Excel.

В ответе укажите числа через пробел в порядке возрастания. Пример ответа: 1 17 39

**Решение.**

Решение приведено на примере csv файла с помощью Excel.

Для начала нужно вставить данные, сделаем это через вкладку “Данные” и вставить данные из файла CSV (рис. 18).

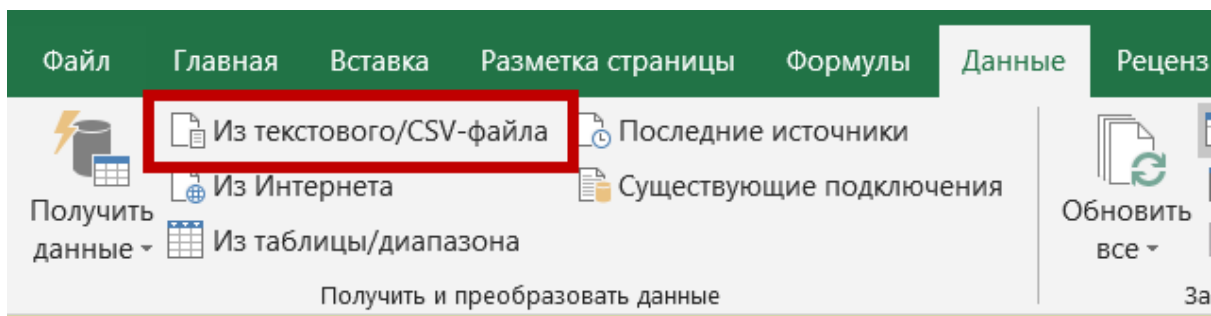


Рис. 18. Загрузка данных в Excel

Далее можно найти среднее значение погоды в городе несколькими способами. Мы воспользуемся сводными таблицами (рис. 19).

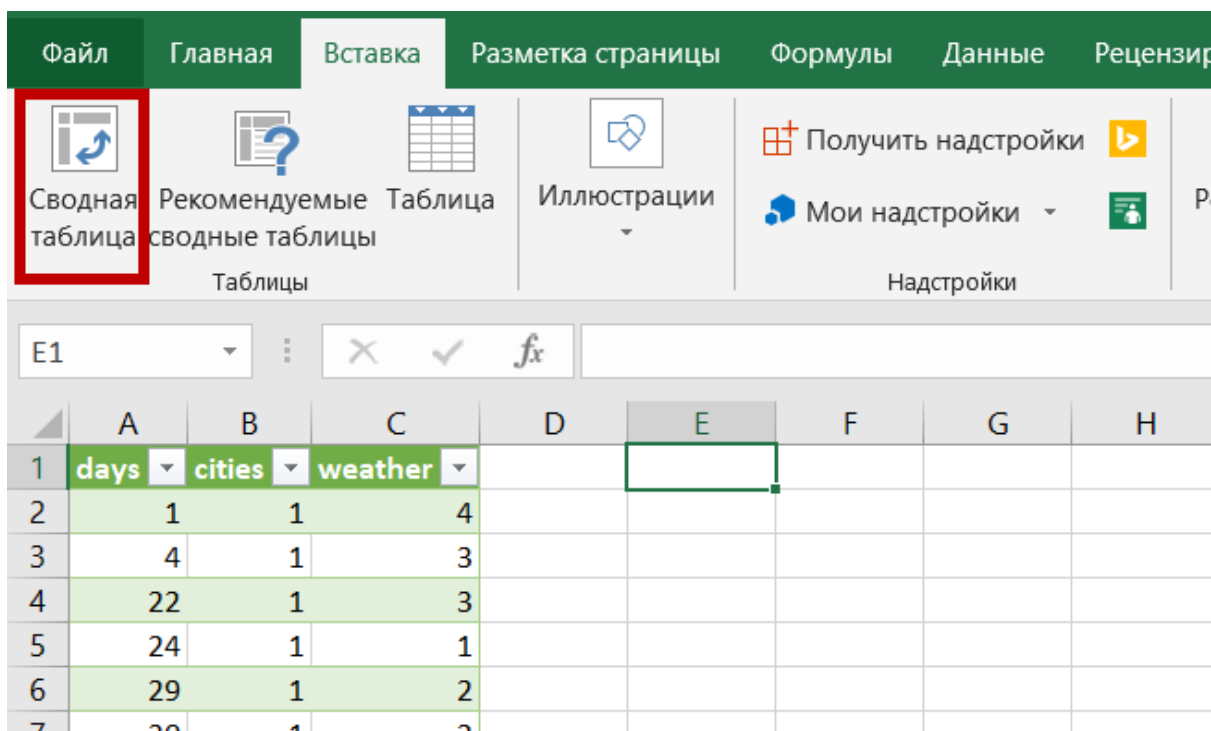


Рис. 19. Создание сводной таблицы в Excel

Настроив сводную таблицу на отображение среднего значения погоды по городам, мы уже можем сравнивать со значением погоды в Энске (рис. 20).

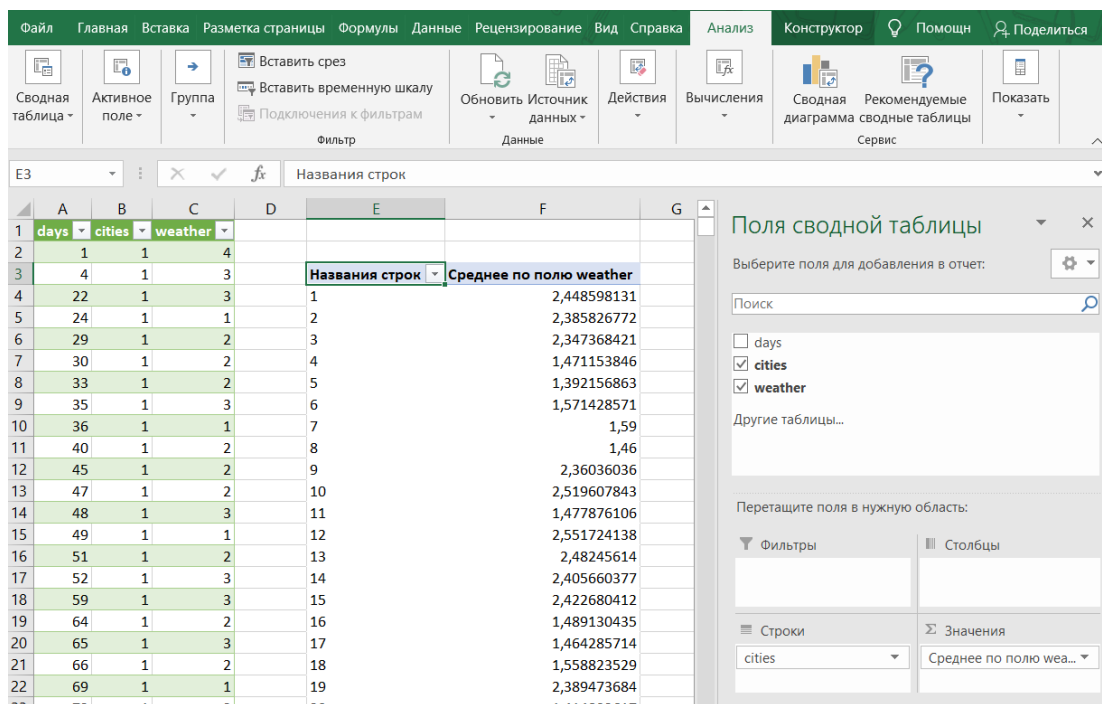


Рис. 20. Средние значения погодных условий в Excel

По условию средняя годовая оценка погодных условий в Энске составляет 2.12 балла, нужно найти максимально близкие значения. Отсортировав среднее погодное значение, можно заметить, что наиболее похожие города по погодным условиям под номерами 3, 9 и 28 (рис. 21).

Названи	Среднее по полю we
22	1,226804124
5	1,392156863
20	1,414893617
23	1,419047619
24	1,428571429
29	1,454545455
8	1,46
17	1,464285714
4	1,471153846
11	1,477876106
16	1,489130435
25	1,540816327
18	1,558823529
21	1,568807339
6	1,571428571
31	1,575221239
7	1,59
3	2,347368421
9	2,36036036
28	2,37254902
2	2,385826772
19	2,389473684
35	2,394230769

Рис. 21. Отсортированные средние значения в Excel

### Пример решения с помощью программы на Python3

```
def ga():
    ga1= float(a[0])
    print(ga1/100)
def kvkm():
    kvkm1= float(a[0])
    print(kvkm1*100)
a=input().split(",")
if a[1] == " кв.км":
    kvkm()
elif a[1] == " га":
    ga()
```

**Ответ:** города под номерами 3,9 и 28 наиболее схожи с Энском



## Задача 2. Единицы измерения площадей земельных участков

### Условие

Измерение площадей городской инфраструктуры обычно осуществляется в гектарах (Га), потому при исследовании аналогов мы будем часто оперировать этой величиной. В школе чаще оперируют понятием кв. километров (кв.км) или кв.метров (кв.м).

Составьте программу, позволяющую осуществлять перевод Га в кв.км. и наоборот (кв.км. в Га).

Входные данные: строка «n, type», где n — число, type — система измерения. В результате: m — число в альтернативной системе измерения.

Пример:

- входные данные: 12.67, кв.км
- результат: 1267.0

Примечание. Выход цифрой, без единицы измерения и без округления

### Решение

#### Вариант решения на Python 3

```
st= input().split(", ")
x=0
if 'кв.км' in st[1]:
    x= float(st[0])*100
elif 'га' in st[1]:
    x = float(st[0])/100
print(x)
```

## Задача 3. Парковые зоны

### Условие

Функциональное зонирование является одной из важнейших составляющих градостроительного регулирования использования территории.

Функциональная зона понимается как территория в определенных границах с однородным функциональным назначением.

Функциональное зонирование парка заключается в рациональном размещении территории зоны отдыха для тех или иных видов деятельности отдыхающего населения. Проектом предусмотрено деление территории на следующие зоны:

- зона тихого отдыха;
- детская игровая зона;
- зона аттракционов;
- зоны физической культуры и спорта;
- культурно-просветительская зона;
- автостоянка для посетителей;
- зона фонтанов и водоемов;
- зона обслуживания;
- хозяйственная зона.

Статистика показывает, что наиболее удачным является следующее соотношение зон.

1. По площади "Зона тихого отдыха" должна быть самой большой.
2. Площади "Детской игровой зоны", "Зоны аттракционов", "Зоны физической культуры и спорта", "Зоны обслуживания" — равны между собой и их сумма в два раза больше "Зоны тихого отдыха".
3. "Зона фонтанов и водоемов" составляет  $\frac{2}{3}$  от "Зоны тихого отдыха".
4. "Культурно-просветительская зона" в 5 раз больше "Хозяйственной зоны".

Ваша задача определить возможные размеры всех зон парка в кв.метрах. Общая парковая зона 15 гектаров (15 Га).

Размер "Автостоянки для посетителей" рассчитывается так: 1 машиноместо на 10 кв.м. Сколько машиномест может быть спроектировано на автостоянке парка?

Примечание. В ответе через пробел перечислите площади в т.кв.м. в следующем порядке:

- зона тихого отдыха;
- детская игровая зона;
- зона аттракционов;
- зоны физической культуры и спорта;
- культурно-просветительская зона;
- автостоянка для посетителей;
- зона фонтанов и водоемов;
- зона обслуживания;
- хозяйственная зона.
- последней цифрой укажите количество машиномест на автостоянке.

Пример ответа: 35 28 15 11 22 10 20 15 50 2000

### Решение

Решений много, но они должны подходить под перечисленные условия, а именно:

- площадь зоны тихого отдыха самая большая.
- Площадь Детской игровой зоны = Площади Зоны аттракционов = Площади Зоны физической культуры и спорта = Площади Зоны обслуживания.
- (Площадь Детской игровой зоны + Площади Зоны аттракционов + Площади Зоны физической культуры и спорта + Площади Зоны обслуживания)/2 = Площади зоны тихого отдыха.
- Площади зоны тихого отдыха/3\*2 = Зона фонтанов и водоемов.

- Культурно-просветительская зона/5 = Площадь хозяйственной зоны
- Сумма всех площадей не превышает 15 гектаров.

Чтобы посчитать количество машиномест нужно площадь автостоянки \* 1000 - чтобы перевести в м<sup>2</sup> и /10 для подсчета мест(исходя из условия на одну машину 10 квадратных метров).

Ответ: возможный вариант ответа: 30 15 15 15 25 10 20 15 5 1000

Диаграмма-ответ к задаче представлена на рис. 22.

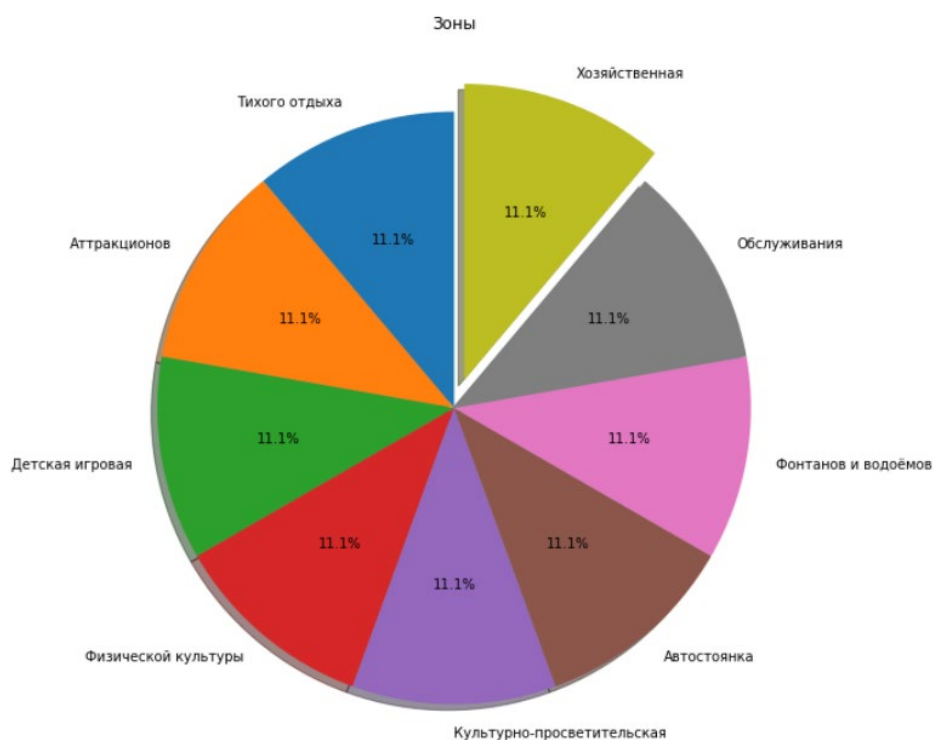


Рис. 22. Диаграмма размеров зон парка

#### Задача 4. Замечательные места

##### Условие.

Поскольку мы ищем готовые решения, то теперь необходимо взять лучшее от парков в тех трех городах, которые, как было показано в задаче 1- "Ищем аналоги", аналогичны Энску. Будем искать в парках этих городов

замечательные места — те, которые притягивают наибольшее количество людей и определим, в чем их "фишки".

Перед стажерами стоит задача написать программу для поиска наиболее посещаемых зон в парках исследуемых городов. Необходимо найти 5 таких зон.

На вход подается массив натуральных чисел  $N$  на  $N$ . В каждой ячейке усредненное количество людей за год. Выведите координаты 5 "замечательных мест" (порядок указания любой).

Примечание. Ответ должен быть перечислением точек координат через запятую и в квадратных скобках.

Пример ответа: [x1, y1], [x2, y2], [x3, y3], [x4, y4], [x5, y5]

### Решение

Предполагается решение в интерактивной тетрадке.

```
import numpy as np
import matplotlib.pyplot as plt
```

```
data = eval(полученные_генерированные_данные_состепика)
maxima = sorted(data.ravel())[:-1][:5]
plt.imshow(data)
points = np.where(np.isin(data, maxima))
plt.scatter(points[1], points[0], color='r');
```

**Ответ.** В коде генерации заложены одинаковые для всех координаты замечательных мест. ответ всегда будет следующим: [2, 31], [5, 4], [10, 12], [35, 49], [48, 30]

Картинка-ответ к задаче представлена на рис. 23.

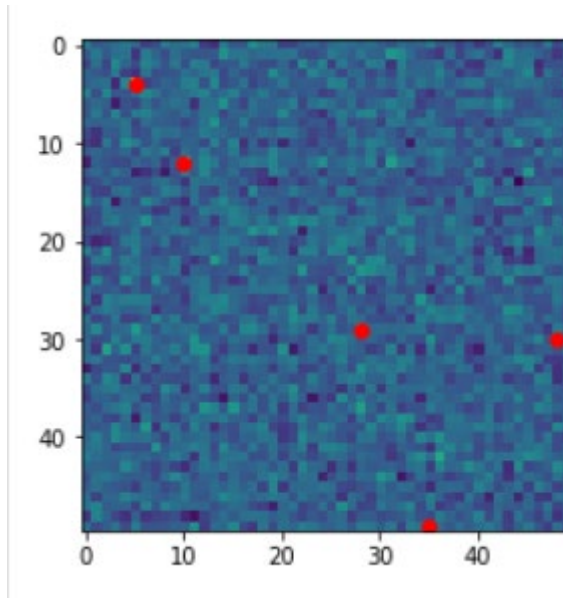


Рис. 23. Посещаемые зоны парка.

### Задача 5. "Фишки" в облаках

#### Условие.

Естественно, что найти готовые успешные решения не так просто. Как узнать, что именно привлекает в людей в замечательные места? В чем их "фишка"? Будем искать наши "фишки" в облаках слов, которые сформируем по текстам, опубликованным в социальных сетях.

У вас есть пять текстов — обобщённых записей, опубликованных в социальных сетях пяти "замечательных мест". К каждому из них постройте облако слов. В ответе к задаче укажите 5 слов — по одному, самому большому слову в облаке. Найдите "фишку"!

#### Текст 1.

Любимая забава детства весной пускать по ручейкам кораблики. Кораблики проплывают под мостом И белыми сверкают парусами, И я радуюсь жизни. Разноцветные кораблики в нашем парке: жёлтые, красные, оранжевые,

— целая флотилия! Все они прилетели сюда по воздуху. Красиво покачиваясь «на крыльях ветерка», кораблики опускались на воду тихо и плавно — без плеска и без шума. Кораблики с поднятыми парусами плывут так важно, горделиво! Мне нравится смотреть, как кораблики раскачиваются на поверхности воды родничков. Вдоль тропинок парка сделаны каналы, так вот в них, пускать кораблики в период наибольшего половодья особенно интересно и так волнительно...

#### Текст 2.

Сначала лёгкое дуновение ветерка всколыхнуло траву и потревожило листву деревьев. Затем с неба посыпались крохотные капельки воды, они ударялись о Барабаны-для-дождя, расставленные на площадке, и вокруг зазвучала тихая мелодичная музыка. Барабаны-для-дождя! Поразили! Стояла рядом зачарованная и чем сильнее становился дождь, тем громче звучали Барабаны-для-дождя. Дети на площадке кружили в танце под музыку, которую исполняли Барабаны-для-дождя.

#### Текст 3.

Песочники нравятся всем. Лепим песочники сами в песочнице, а потом забираем в пекарне и едим яблочные, вишневые, клубничные песочники. Песочники - готовят как сладкие, так и не сладкие. Дети в восторге от песочников. Играли в песочники: сначала лепили их из влажного песка, а потом по нашему дизайну в пекарне испекли яблочный и творожный песочные пироги. Здорово придумано! Пироги, в которых используется песочное тесто — это песочники. Пироги из песка - песочники. Песочников много не бывает. Дети налепили 10 штук — сколько мы их будем есть? Песочники — детская мечта, воплощенная в реальность.

#### Текст 4.

Как я люблю ForrestCafe! ForrestCafe— это уголок волшебного леса, в котором каждый может вспомнить детство и погрузиться в чарующую сказочную атмосферу. Горожане! Слушайте, в Forrest Cafe можно вкусно пообедать под сенью фантастического дуба и живых папоротников, послушать концерты музыкальных групп и потанцевать. Как же здесь классно! Фантастический интерьер ForrestCafe воплощает мои представления об антураже дремучего леса из русской народной сказки. Два небольших зала ForestCafe, разделенных сценой, утопают под сенью могучих дубов, вырастающих прямо из пола. Панели из резного дерева со структурой растрескавшейся коры, переплетение ветвей, расползающихся по потолку сложной формы с необычными светильникам в виде цветочных лепестков, зелень в горшках, красочный аквариум погружают гостей в загадочную атмосферу векового лесного сумрака и покоя. Как же всей нашей семье нравится ForrestCafe! В ForrestCafe Столики и лавки из темного дерева в фантастическом стиле «избушки бабы Яги», барная стойка в том же стиле, украшенная резьбой в виде растительного орнамента, картины в рамках из веток, резные фигурки фантастических животных на стенах, расписные абажуры ламп над столиками объединяются в целостную композицию, в которой уютно и интересно и детям, и взрослым. ForrestCafe отличное место для отдыха всей семьей! Как в гостях у сказки побывали! А музыка дождя рядом придают этому месту еще большую «ламповость».

#### Текст 5.

Прекрасный музей современного искусства! Хорошая выставка и приятный персонал. Музей находится в парке, что очень удобно. После выставки можно прогуляться в красивом месте. Классно! Очень понравилось! Посещал данный музей. Атмосфера понравилась. Что написать про современное искусство...есть моменты, которые не укладываются в



сознании...но искусство каждый воспринимает по-своему...очень запомнилось. Музей современного искусства хорошо продуман в архитектурном плане (внешний вид здания, интерьеры, туалеты и разные мелочи, вроде номерков в гардеробе. Всегда приятно прийти в этот музей. С точки зрения архитектуры музей имеет очень красивое. Просторный современный музей. Пространство соответствует статусу. Экспозиция подобрана гармонично Музей впечатляет! Неожиданно организовано пространство, стильно, очень много воздуха. Не могу назвать себя знатоком современного искусства, и уж точно не разбираюсь в постмодернистских философских течения, но равнодушным не оставляет точно. Музей "МУ" прекрасен! Совместить поход в него можно с прогулкой по парку. Удобно и познавательно! Музей насыщен событиями: лекции, кинопросмотры, встречи и пр. Своеобразный музей. Я думаю, что хотя бы разок его следует посетить всем. Действительно современное здание, напичканное всеми элементами современной музейной электроники. Удобно, что он расположен в парке, после посещения можно прогуляться, посетить кафе.

Примечание. В ответе перечислите 5 слов

### Решение

Чтобы создать облако слов, можно воспользоваться любым сайтом, который предоставляет данные услуги, или по примеру из заданий первого тура и заданий на самоподготовку воспользоваться интерактивной тетрадкой.

### Примеррешениявтетрадке.

```
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
from collections import Counter
import matplotlib.pyplot as plt
comments_1 = ""
```

Любимая забава детства весной пускать по ручейкам кораблики. Кораблики проплывают под мостом И белыми сверкают парусами, И я радуюсь жизни. Разноцветные кораблики в нашем парке: жёлтые, красные, оранжевые, — целая флотилия! Все они прилетели сюда по воздуху. Красиво покачиваясь «на крыльях ветерка», кораблики опускались на воду тихо и плавно — без плеска и

без шума. Кораблики с поднятыми парусами плывут так важно, горделиво! Мне нравится смотреть, как кораблики раскачиваются на поверхности воды родничков. Вдоль тропинок парка сделаны каналы, так вот в них, пускать кораблики в период наибольшего половодья особенно интересно и так волнительно...

""

comments\_2 = ""

Сначала лёгкое дуновение ветерка всколыхнуло траву и потревожило листву деревьев. Затем с неба посыпались крохотные капельки воды, они ударились о Барабаны-для-дождя, расставленные на площадке, и вокруг зазвучала тихая мелодичная музыка. Барабаны-для-дождя! Поразили! Стояла рядом зачарованная и чем сильнее становился дождь, тем громче звучали Барабаны-для-дождя. Дети на площадке кружили в танце под музыку, которую исполняли Барабаны-для-дождя.

""

comments\_3 = ""

Песочники нравятся всем. Лепим песочники сами в песочнице, а потом забираем в пекарне и едим яблочные, вишневые, клубничные песочники. Песочники - готовят как сладкие, так и не сладкие. Дети в восторге от песочников. Играли в песочники: сначала лепили их из влажного песка, а потом по нашему дизайну в пекарне испекли яблочный и творожный песочные пироги. Здорово придумано! Пироги, в которых используется песочное тесто – это песочники. Пироги из песка - песочники. Песочников много не бывает. Дети налепили 10 штук – сколько мы их будем есть? Песочники – детская мечта, воплощенная в реальность.

""

comments\_4 = ""

Как я люблю ForrestCafe! ForrestCafe- это уголок волшебного леса, в котором каждый может вспомнить детство и погрузиться в чарующую сказочную атмосферу. Горожане! Слушайте, в Forrest Cafe можно вкусно пообедать под сенью фантастического дуба и живых папоротников, послушать концерты музыкальных групп и потанцевать. Как же здесь классно! Фантастический интерьер ForrestCafe воплощает мои представления об антураже дремучего леса из русской народной сказки. Два небольших зала ForestCafe, разделенных сценой, утопают под сенью могучих дубов, вырастающих прямо из пола. Панели из резного дерева со структурой растрескавшейся коры, переплетение ветвей, расплзающихся по потолку сложной формы с необычными светильникам в виде цветочных лепестков, зелень в горшках, красочный аквариум погружают гостей в загадочную атмосферу векового лесного сумрака и покоя. Как же всей нашей семье нравится ForrestCafe! В ForrestCafe Столики и лавки из темного дерева в фантастическом стиле «избушки бабы Яги», барная

стойка в том же стиле, украшенная резьбой в виде растительного орнамента, картины в рамках из веток, резные фигурки фантастических животных на стенах, расписные абажуры ламп над столиками объединяются в целостную композицию, в которой уютно и интересно и детям, и взрослым. ForrestCafe отличное место для отдыха всей семьей! Как в гостях у сказки побывали! А музыка дождя рядом придает этому месту еще большую ламповость.

""""

```
comments_5 = """"
```

Прекрасный музей современного искусства! Хорошая выставка и приятный персонал. Музей находится в парке, что очень удобно. После выставки можно прогуляться в красивом месте. Классно! Очень понравилось! Посещал данный музей. Атмосфера понравилась. Что написать про современное искусство...есть моменты, которые не укладываются в сознании...но искусство каждый воспринимает по своему...очень запомнилось. Музей современного искусства хорошо продуман в архитектурном плане (внешний вид здания, интерьеры, туалеты и разные мелочи, вроде номерков в гардеробе. Всегда приятно прийти в этот музей. С точки зрения архитектуры музей имеет очень красивое. Просторный современный музей. Пространство соответствует статусу. Экспозиция подобрана гармонично Музей впечатляет! Неожиданно организовано пространство, стильно, очень много воздуха. Не могу назвать себя знатоком современного искусства, и уж точно не разбираюсь в постмодернистских философских течения, но равнодушным не оставляет точно. Музей "МУ" прекрасен! Совместить поход в него можно с прогулкой по парку. Удобно и познавательно! Музей насыщен событиями: лекции, кинопросмотры, встречи и пр. Своеобразный музей. Я думаю, что хотя бы разок его следует посетить всем. Действительно современное здание, напичканное всеми элементами современной музейной электроники. Удобно, что он расположен в парке, после посещения можно прогуляться, посетить кафе.

""""

**#Выделяем слова которые точно не должны использоваться при построении облака слов!**

```
stopwords = set(STOPWORDS)
stopwords.update(["на", "все", "эта", "не", "вчера", "уже"])
#Создаем облако
wordcloud = WordCloud(stopwords = stopwords,
background_color="white").generate(comments_1)
# Рисуемрезультат
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.show()
#Создаемоблако
```

```
wordcloud = WordCloud(stopwords = stopwords,  
background_color="white").generate(comments_2)  
# Рисуемрезультат  
plt.imshow(wordcloud, interpolation="bilinear")  
plt.axis("off")  
plt.show()  
#Создаемоблако  
wordcloud = WordCloud(stopwords = stopwords,  
background_color="white").generate(comments_3)  
# Рисуемрезультат  
plt.imshow(wordcloud, interpolation="bilinear")  
plt.axis("off")  
plt.show()  
#Создаемоблако  
wordcloud = WordCloud(stopwords = stopwords,  
background_color="white").generate(comments_4)  
# Рисуемрезультат  
plt.imshow(wordcloud, interpolation="bilinear")  
plt.axis("off")  
plt.show()  
#Создаемоблако  
wordcloud = WordCloud(stopwords = stopwords,  
background_color="white").generate(comments_5)  
# Рисуемрезультат  
plt.imshow(wordcloud, interpolation="bilinear")  
plt.axis("off")  
plt.show()
```

## Ответ.

Исходя из получившихся облаков-слов можно выделить следующие слова:

- кораблики;
- барабаны или барабаны дождя (и тот и другой вариант засчитывается);
- песочники;
- ForrestCafe(принимаются ответы и с любым одним словом из данной картинки);





Рис. 24. Облака слов

## Задача 6. Предпочтения

### Условие

Полностью копировать готовые решения — дело неблагодарное, поэтому к позаимствованным "фишкам" для "замечательных мест" необходимо добавлять что-то востребованное именно в Энске.

По графику предпочтений горожан (рис. 25) выберите два объекта, которые стоит добавить в проект при реконструкции парка.

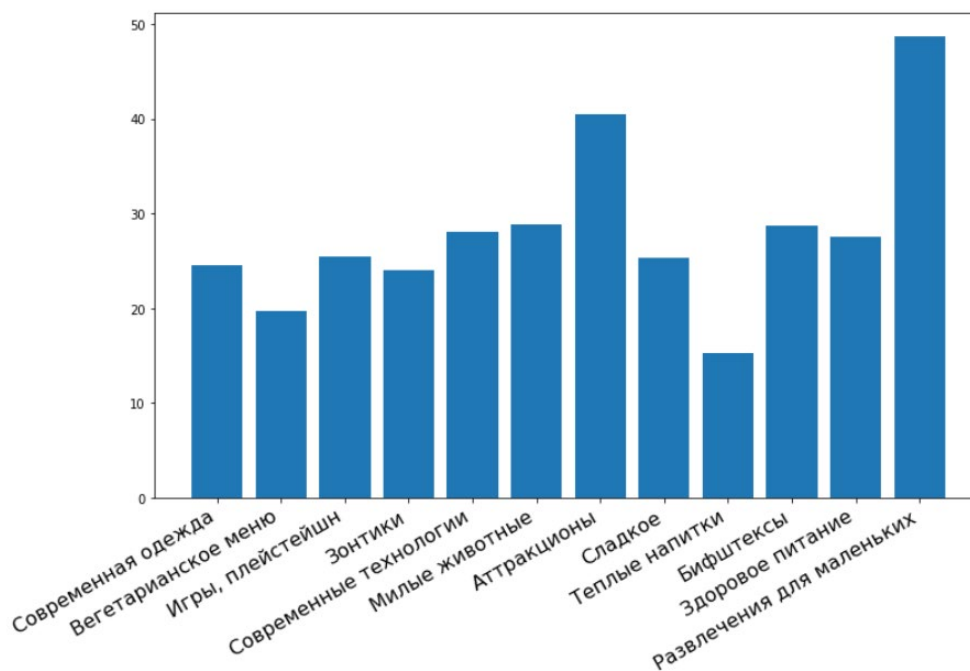


Рис. 25. График предпочтений горожан

Предполагаемые для строительства объекты:

- Колесо обозрения. Окна в каютах, при соприкосновении стекла с каплями дождя – небо в окнах становится радужным.
- Гриль-бар. Горячие закуски.
- Детский парк под открытым небом (но застекленный).
- Музыкальная сцена.
- Новые физические тренажёры X-FIT.
- Подземная парковка.

## Решение

Так как погода в Энске чаще всего пасмурная и дождливая (исходя из задач первого тура), то текущие предложения аттракционов более актуальны, чем бывшие аттракционы на крыше, которые большую часть времени

простаивали бы и ржавели. И так как горожане по-прежнему заинтересованы именно в них, то следует принять заявку Колеса обозрения.

Так же, горожане предпочли бы Развлечения для маленьких, так как такое предложение одно, к тому же, застекленному парку не помешает погода, то эту заявку тоже следует принять.

#### **Ответ.**

Стоит добавить в проект:

- Детский парк под открытым небом (но застекленный).
- Колесо обозрения. Окна в каютах при соприкосновении стекла с каплями дождя небо в окнах становится радужными.

### **Задача 7. Удобство в дорогах**

#### **Условие**

Итак, junior-стажеры успешно разобрались с задачей зонирования и "замечательных" мест, нашли "фишки" в решениях для парков городов аналогичных Энску. Теперь важно продумать пешеходную логистику парковой территории (рис. 26).



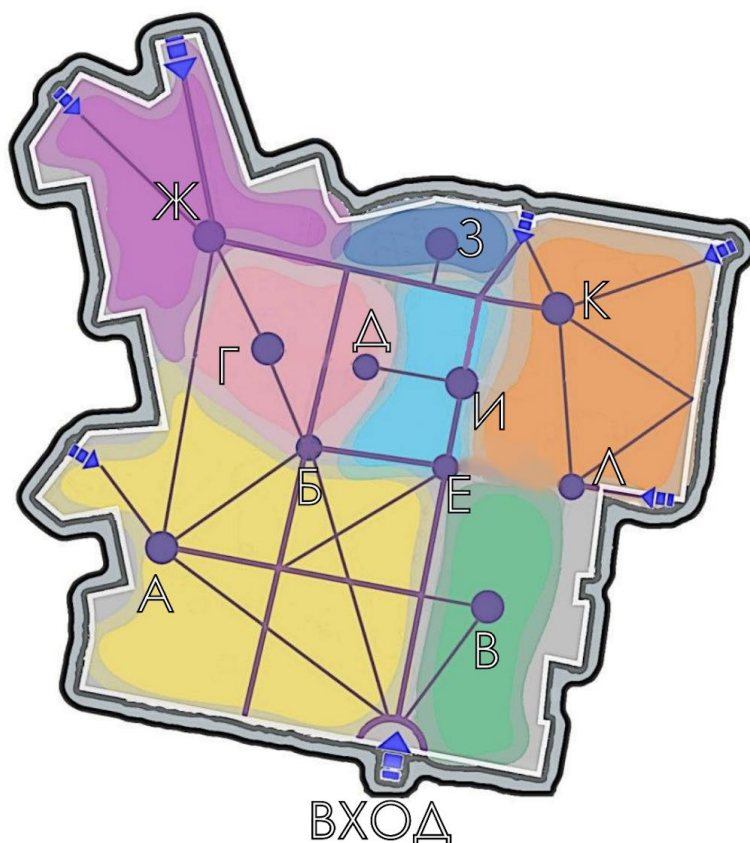


Рис. 26. Дорожки парковой территории

Рассмотрите внимательно карту-схему территории парка. Каждая точка карты-схемы проектируется как место для размещения определенного паркового объекта. Парковый объект будет посещаемым, если точка является доступной.

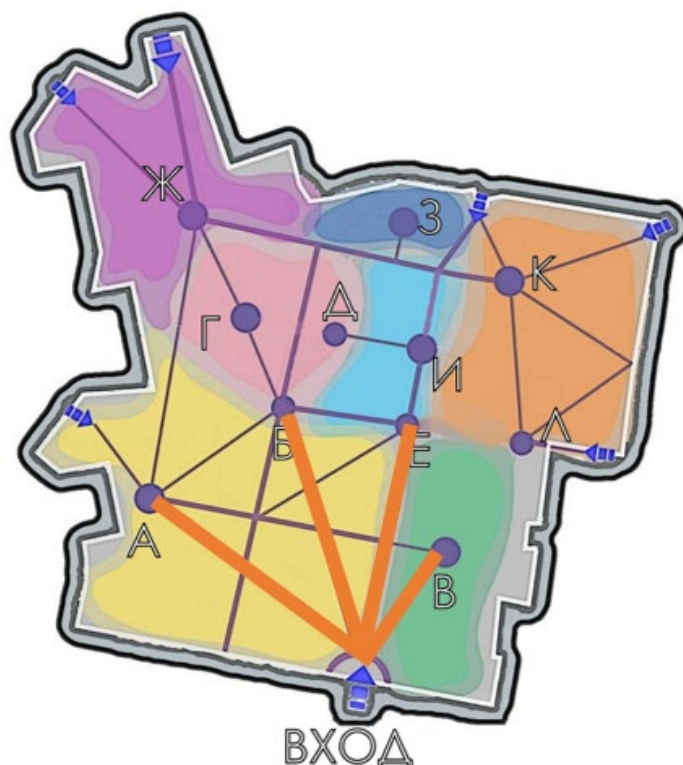
Мы считаем точку доступной, когда добраться до нее от центрального входа можно, минуя не более одного объекта. Таким образом, каждая первая и вторая точка на какой-либо тропинке от центрального входа — доступна, но если путь к интересующему нас объекту лежит через два других, то объект располагается за пределами доступности.

Junior-стажерам необходимо найти такие точки, которые не отвечают вышеприведенному условию доступности.

Примечание. Перекресток точкой не считается! Обратите внимание, что центральный вход только один. На схеме он подписан. Все точки обозначены русскими буквами.

## Решение

Прочертим линии к точкам, до которых мы можем сразу добраться от центрального входа (рис. 27).



*Рис. 27. Ближайшие ко входу точки парка.*

Точки А, Б, Е, В очевидно, доступны.

Теперь прочертим дороги к точкам “второго уровня”, к которым мы можем добраться через точно доступные точки и которые по условию тоже являются доступными (рис 28).

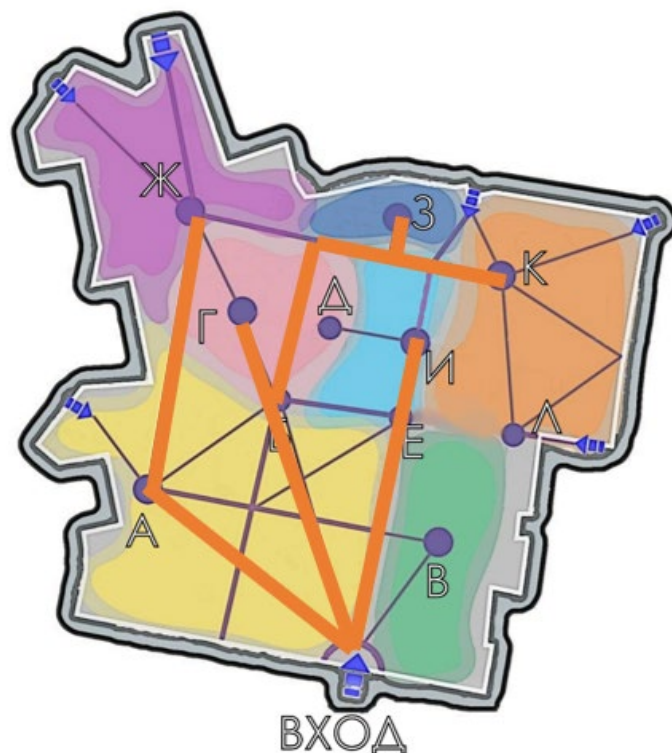


Рис. 28. Второй уровень доступных точек парка

Если внимательно посмотреть, то точки З и К, тоже доступны, так как перекрестки точками не считаются. Из этой картинки следует, что точки Г, Ж, З, И, К - тоже доступны.

Все другие точки будут считаться недоступными, так как придется миновать уже два объекта, а в условии сказано минуя не более одного объекта. Соответственно точки Д и Л - недоступны.

**Ответ**

Точки Д и Л располагаются за пределами доступности

### Задача 8. Центр.

**Условие.**

Любое место должно быть центрировано — это позволяет быстрее ориентироваться.

Центрированное место — это точка, из которой до других точек выходит наибольшее количество быстрых маршрутов (таких маршрутов, в которых все точки удовлетворяют условию доступности). Иначе, от центра любой объект находится в доступной точке. Схема парка представлена на рис. 26.

Ваша задача найти такой центр. В ответе напишите название точки-центра парка (русская буква).

### Решение

Из точки Б напрямую доступны 7 других точек: А,В,Г,Е,Ж,З,К. Из других точек напрямую доступно меньшее количество точек.

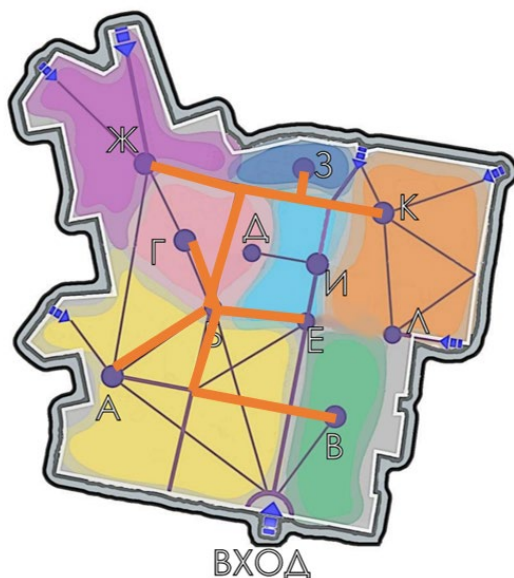


Рис. 29. Маршруты из точки Б

### Ответ

Центр исследуемого объекта - точка Б

## Задача 9. Центральная достопримечательность

### Условие

Теперь следует выбрать центральную "фишку", тот объект, который станет главной достопримечательностью парка "Роднички".

Мы опросили людей, чтобы они оценили каждую достопримечательность от 0 до 10. Данные опроса были занесены в таблицу. Таблица также есть в формате csv и pythondict.

Ваша задача построить график усредненных значений оценки достопримечательностей по данным из любого файла.

В ответ напишите достопримечательность с самым высоким баллом.

### Решение

Предполагается решение на интерактивной тетрадке

```
from random import shuffle
import pandas as pd
data = (данные из файла со степика)
data.mean()
```

В результате выведется следующее:

Детский парк	5.488977
Кораблики	6.291124
Барабаны-для-дождя	6.671595
Лесное кафе	6.810179
МУ	6.891329
Песочники	7.035429
Колесо обозрения	7.337698

### Ответ

Данные в файле сгенерированы так, чтобы самый высокий балл был у Колеса обозрения. Поэтому ответ - Колесо обозрения

## Задача 10. Время работы

### Условие

Как и в предыдущий раз, нам важно определиться со временем работы парка. Будем исходить из того, что оптимальное время работы вновь 8 часов. Только теперь нам важнее не количество людей в определенных центрах города (как было в случае развлекательного центра), а количество людей разного возраста рядом с парком. Помните — парк молодежный. Поэтому самое главное, чтобы время устраивало именно эту возрастную категорию.

Ниже приведено три графика активности для разных возрастных категорий: с 5 до 24 лет (рис. 30), с 24 до 38 (рис. 31) и на третьем людей старше 38 (рис. 32). Исследуемая возрастная категория (5–24, 24–38). Определите по графикам оптимальное время работы парка.

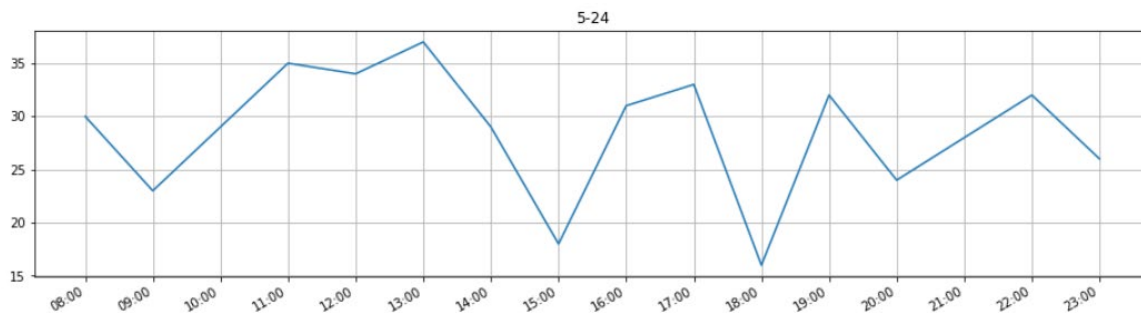


Рис. 30. График активности людей 5–24 лет

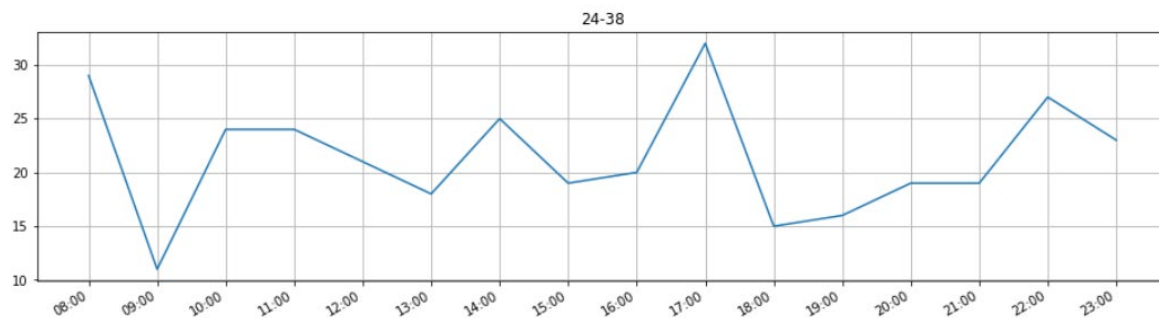


Рис. 31. График активности людей 24–38 лет

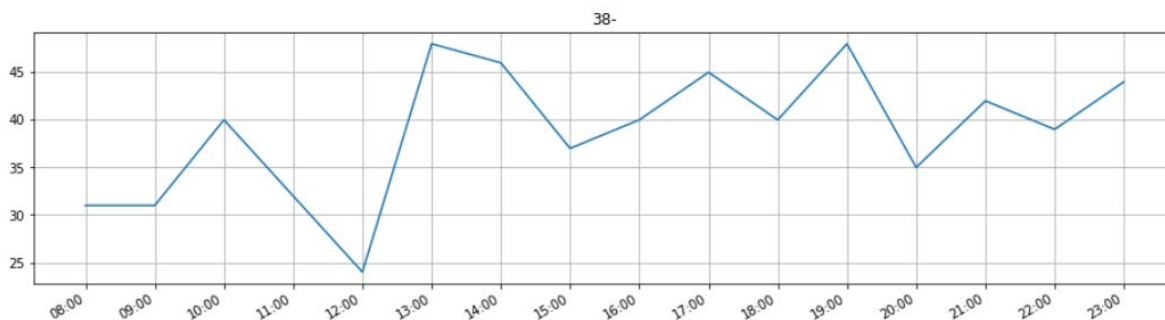


Рис. 32. График активности людей старше 38 лет

**Примечание.** Напишите ответ слитным четырехзначным числом. Например, с 9 утра до 21 вечера - 0921.

### Решение

Имеет смысл использовать только графики первых двух категорий, так как наша исследуемая аудитория - молодежь отображена только в тех двух графиках. Посмотрев на них, можно заметить два пика: 13:00 и 17:00 - в это время парк точно должен быть открыт. Общее время работы парка 8 часов. Период с 13 до 17 — это 4 часа.

Но после 17:00 идет резкое падение количества молодежи на обоих графиках (рис. 33).

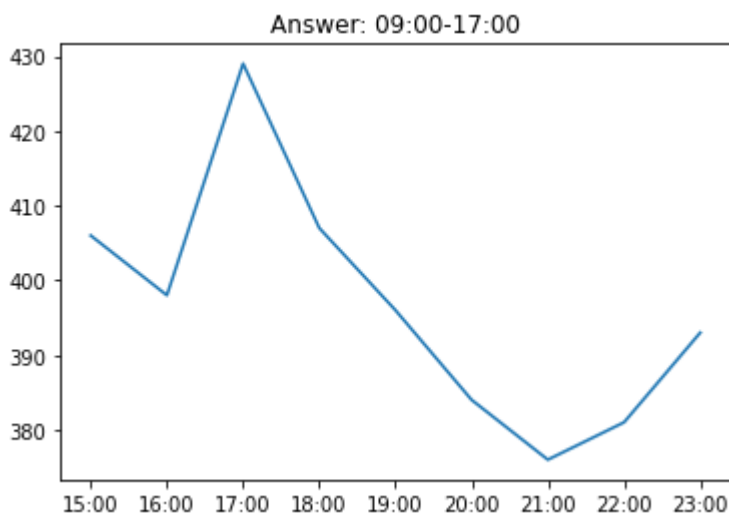


Рис. 33. Посещаемость парка после 15:00

Из графика следует, что оптимальное время работы парка с 9 утра до 17 вечера.

Ответ. Оптимальное время работы с 9 утра до 17 вечера.

### **Третий модуль индивидуального этапа.**

#### **Постановка проблемы для участников.**

Город Энск стремится к процветанию и старается предложить своим жителям и гостям лучшие решения для благоприятного пребывания в нем! Но впереди, Junior-стажеры, вас ждут еще 10 заданий по социальному исследованию территории района "Эн" (рис. 34) на периферии города "Энск". Исследование социальных активностей, поведения поможет выявить потребности жителей района, их предпочтения, укажет на проблемы.

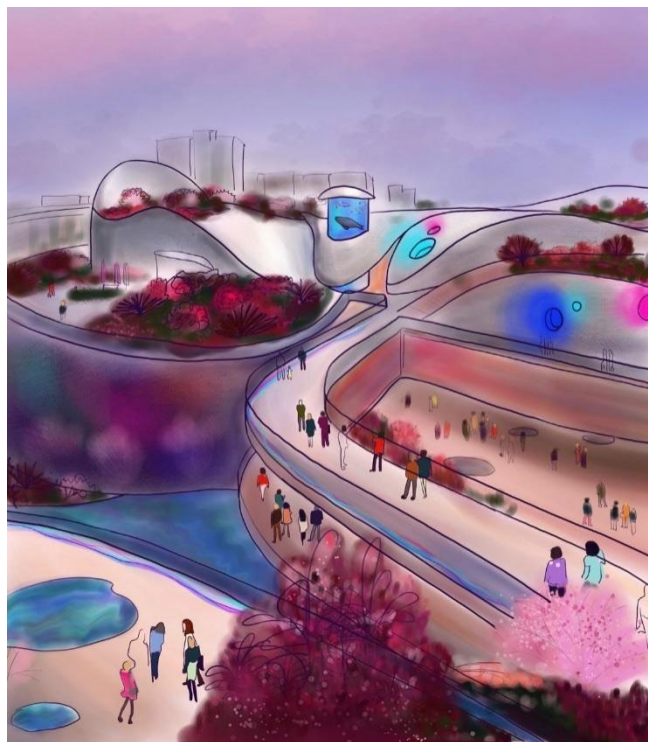


Рис. 34. Район Эн, города Энск.



### Задача 1. Сбор данных для исследований

Ниже представлен скриншот страницы сайта города Энск (рис. 35), где собраны отзывы о знаковых местах района Эн. Так сайт выглядит для пользователей: галерея фото с привязкой к местам, описанием к ним и лайками.

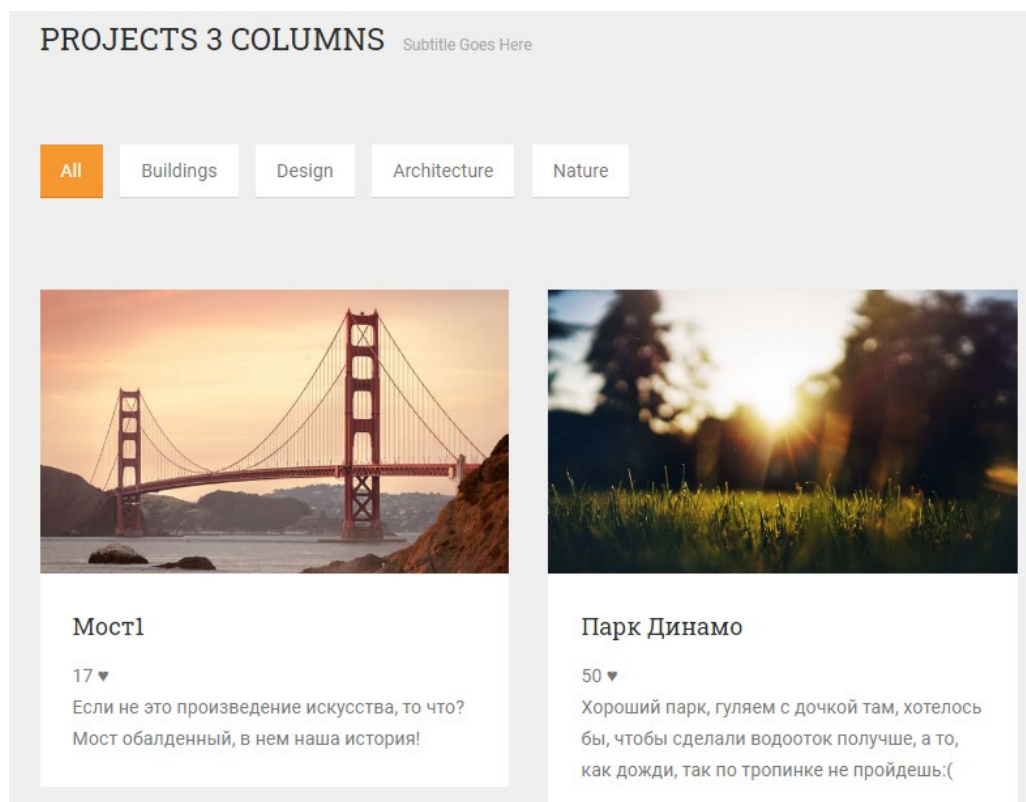


Рис. 35. Фрагмент страницы сайта г. Энск.

Такие сайты - с оценками и мнением пользователей становятся значимыми источниками информации для исследований самых разных объектов, в том числе и для формирования мнения относительно инфраструктуры городских территорий. Переносить данные для анализа с сайта руками достаточно трудоемкий и бессмысленный процесс. Сбор данных с интернет-ресурса можно автоматизировать, составив специальные программы - парсеры. Задача это сложная. на первом этапе junior-стажерам предстоит разобраться с кодом страницы сайта, найти в коде "метки" (теги), которые будут указывать парсеру, какие данные и откуда считывать.

Страница сайта написана на языке разметки html. Ниже представлен фрагмент кода.

```
<!DOCTYPE html>
<html>
<head>
  <title>Места района города Энск</title>
  <meta charset="utf8">
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <div class="col-md-4 project-item mix buildings">
    <div class="project-thumb">
      <imgsrc="images/projects/project_1.jpg" alt="">
      <div class="overlay-b">
        <div class="overlay-inner">
          <a href="images/projects/project_1.jpg" class="fancybox fa fa-
expand" title="Project Title Here"></a>
        </div>
      </div>
    </div>
    <div class="box-content project-detail">
      <h2>Мост1</h2>
      <p>Мост обалденный, в нем наша история! Величественная
постройка, отражает прошлое нашего района!</p>
      <span>17 ♥</span>
    </div>
  </div>
  <div class="col-md-4 project-item mix nature">
    <div class="project-thumb">
      <imgsrc="images/projects/project_2.jpg" alt="">
      <div class="overlay-b">
        <div class="overlay-inner">
          <a href="images/projects/project_2.jpg" class="fancybox fa fa-
expand" title="Project Title Here"></a>
        </div>
      </div>
    </div>
    <div class="box-content project-detail">
      <h2>Парк Динамо</h2>
```

```
<p>Хороший парк, гуляем с дочкой, но после дождей тропики  
размывает. Природа замечательное, белочек можно покормить!</p>  
<span>50 ♥</span>  
</div>  
</div>  
</body>  
</html>
```

Код html представляет собой вложенную структуру различных тегов, заключенных в треугольные скобки <>. Как правило теги парные, внутри них заключена разнообразная информация. Нам, для того чтобы приступить к анализу данных, необходимо найти в разметке страницы определенные теги и извлечь информацию, которую они содержат.

В коде данного сайта представлены следующие теги: ['div', 'h2', 'link', 'head', 'title', 'span', 'img', 'body', 'meta', 'html', 'p', 'a']

Например, чтобы получить количество лайков ("17 ♥" и "50 ♥") - нужно найти тег span: 17 ♥.

Определите, в каких тегах содержится информация (данные):

- Название места (локации) района, в котором сделана фотография.
- Описание к фотографии.

Примечание. Введите названия тегов (пишутся английскими буквами)

## Решение

При рассмотрении структуры документа нужно найти название места. Из картинки (рис. 35) видно, что нам нужна надпись “Мост1” и “Парк Динамо”. Внимательно просмотрев документ, находим следующие строки:

```
<h2>Мост1</h2>  
<h2>Парк Динамо</h2>
```

Других надписей с содержанием “Мост1” и “Парк Динамо” нет, значит тег, в котором содержатся названия мест - h2.

Описание к фотографии. Опять же, посмотрев на картинку (рис. 35), находим эти описания. В коде легко заметить строки с ними.  
<p>Мост обалденный, в нем наша история! Величественная постройка, отражает прошлое нашего района!</p>  
<p>Хороший парк, гуляем с дочкой, но после дождей тропики размывает. Природа замечательное, белочек можно покормить!</p>

Других подобных надписей нет, соответственно тег для описания фото – р

**Ответ:** название места: h2, описание к фото: р.

## Задача 2. Анализ отзывов

С помощью сбора данных (задача №1) компания AltFuture получила готовый dataset с отзывами пользователей: их мнением об исследуемой территории. Теперь junior -стажеры должны обработать обобщенный текст и выделить в нем ключевое слово, то которое чаще всего используют пользователи, описывая исследуемый район Энска.

Сгенерируйте облако слов по всем отзывам, предварительно исключив из списка слова: ["на", "все", "эта", "не", "уже", "это", "что", "по", "так", "но"].

### Отзывы.

['Мост обалденный, в нем наша история! Величественная постройка, отражает прошлое нашего района!', 'Хороший парк, гуляем с дочкой, но после дождей тропики размывает. Природа замечательное, белочек можно покормить!', 'Парк к сожалению старый, но в этом его плюс. Прошла целая эпоха, история, всегда навевает прошлое этого места. Хорошо здесь гулять, подумать, жаль, что стоя, хочется лавочки', 'Как хорошо гулять по аллее на закате. Природа супер! #настроениеосень, погода сегодня была замечательная!', 'Обожаю тусовки с друзьями летом на речке, это наше место, природа... Благодать!)) хд', 'Сегодня

было чудесное настроение, но я приехал на работу и офигел! Так много мусора на дорогах (ветер разнес, у нас что, дворники совсем не работают???, 'Тут можно покормить белочек. Люблю это место, как здорово, что природа рядом с нами', 'Пробки просто кошмар! Невозможно доехать до центра за 30 минут, приходится стоять долго ((', 'Здесь я могу ни о чем не думать, читать книгу, наслаждаться жизнью, прекрасное место', 'Вы замечали какая культура у наших людей? Опять не убрали мусор за собой (Жалко лес', 'Людей нет, тишина! Благодать! Играли в фрисби с собакой, кайф']

В ответе укажите слово, которое чаще упоминается

### Решение

Для нахождения ответа к данной задаче нужно построить облако слов. Можно использовать сторонний сервис или интерактивную тетрадку с аналогичной задачей из набора пробных задач.

*Пример решения с использованием интерактивной тетрадки*

comments = данныеизусловия

```
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
from collections import Counter
import matplotlib.pyplot as plt
#Выделяем слова которые точно не должны использоваться при построении
облака слов!
stopwords = set(STOPWORDS)
stopwords.update(["на", "все", "эта", "не", "уже", "это", "что", "по", "так", "но"])

#Создаем облако
wordcloud = WordCloud(stopwords = stopwords,
background_color="white").generate(comments)

# Рисуем результат
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```



Рис. 36. Результат работы программы – облако слов.

Ответ: чаще всего упоминается слово “Природа”

### Задача 3. Анализ целевой аудитории района

#### Условие

Администрацией района Эн проводился опрос населения. В ходе опроса участники, должны были выбирать возрастную категорию аудитории, которую они представляют: студент, пенсионер, работающее население. Кроме того, респондентов просили отметить, есть ли у них дети. Компания AltFuture просмотрела результаты опроса и сформулировала для своих стажеров новую задачу.

Результаты опроса собраны в два массива:

- groups — группа (категория), к которой участники себя относили во время опроса ("Студент", "Пенсионер" или "Работаю");
- child — есть ли у них дети (True — есть, False — нет).

Аналитики компании считают, что из массива данных категории респондентов целесообразно пересобрать в группы: "молодежь", "родители с детьми", "пенсионеры" и "другое".

Junior-стажерам предстоит проанализировать данные и построить диаграмму-пирог со следующими категориями:

- Родители с детьми, если: groups='Работаю' или groups ='Студент' и есть дети;
- Молодежь: groups ='Студент' и детей нет;
- Пенсионеры: groups='Пенсионеры';
- Другое groups="Работаю" и детей нет.

Определите, какова доля участников каждой категории в опросе. И расположите группы в порядке убывания их количества: от самой большой группы к самой маленькой.

### Данные.

```
groups = ['Работаю', 'Студент', 'Работаю', 'Студент', 'Работаю', 'Пенсионер',  
'Работаю', 'Студент', 'Студент', 'Пенсионер', 'Работаю', 'Работаю', 'Пенсионер',  
'Пенсионер', 'Студент', 'Работаю', 'Работаю', 'Студент', 'Работаю', 'Работаю',  
'Работаю', 'Работаю', 'Работаю', 'Студент', 'Студент', 'Студент', 'Работаю',  
'Студент', 'Пенсионер', 'Студент', 'Работаю', 'Работаю', 'Работаю', 'Студент',  
'Студент', 'Работаю', 'Работаю', 'Студент', 'Работаю', 'Студент', 'Студент',  
'Работаю', 'Студент', 'Работаю', 'Работаю', 'Студент', 'Студент', 'Работаю',  
'Работаю', 'Работаю']
```

```
child = [False, True, False, True, False, True, False, True, True, False, False, False,  
False, True, True, True, True, True, True, True, False, False, False, False, True,  
False, False, False, True, False, False, False, True, True, True, False, False, True,  
False, False, False, True, False, True, False, False, False, False, False]
```

### Решение

*Возможное решение с использованием интерактивной тетрадки*

```
groups = данные из условия
```

```
child = данные из условия
```

```
category = [0,0,0,0]
```

```
i = 0
```

```
while(i<len(groups)):
```

```
    if (groups[i] == 'Работаю' or groups[i] == 'Студент') and child[i]:
```

```
        category[0] += 1
```

```
    elif (groups[i] == 'Студент') and not child[i]:
```

```

category[1]+=1
elif (groups[i] == 'Работаю') andnot child[i]:
    category[2]+=1
else:
    category[3]+=1
i+=1
labels = ['Родители с детьми', 'Молодежь', "Другое", "Пенсионеры"]

import matplotlib.pyplot as plt
plt.figure(figsize = (10,10))
plt.pie(category, labels=labels, shadow=True, startangle=90)
plt.show()

```

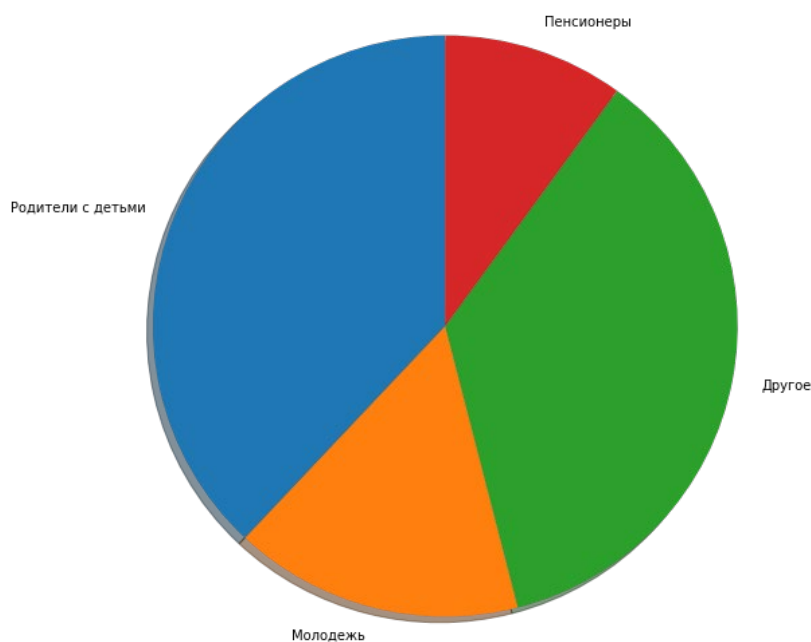


Рис. 37. Диаграмма-пирог численности разных категорий населения

На диаграмме видно, что “Родители с детьми” больше всего, далее следует категория “Другое”, затем “Молодежь” и последние по количеству “Пенсионеры”

Ответ: Родители с детьми, Другое, Молодежь и Пенсионеры

#### Задача 4. Какие в районе проблемы?

Условие.



Вам предоставляются данные, представляющие из себя список проблем данного района, выявленных в результате опросов, интервью и данных с интернет-ресурса города.

Вам необходимо найти 2 главные проблемы территории.

Пример: ['Мусор', 'Безопасность', 'Шум', 'Несанкционированная реклама', 'Шум', 'Шум', 'Транспорт', 'Мусор', 'Мусор', 'Мусор', 'Скучно', 'Скучно', 'Мусор', 'Скучно', 'Несанкционированная реклама']

Ответ: Мусор, Шум и Скучно (если проблемы имеют равное значение, то указываем их все).

#### **Данные для анализа.**

['Мусор', 'Безопасность', 'Шум', 'Несанкционированная реклама', 'Шум', 'Шум', 'Транспорт', 'Мусор', 'Мусор', 'Мусор', 'Скучно', 'Скучно', 'Мусор', 'Скучно', 'Несанкционированная реклама', 'Безопасность', 'Скучно', 'Мусор', 'Несанкционированная реклама', 'Мусор', 'Скучно', 'Скучно', 'Безопасность', 'Безопасность', 'Безопасность', 'Мусор', 'Шум', 'Шум', 'Шум', 'Несанкционированная реклама', 'Мусор', 'Мусор', 'Мусор', 'Безопасность', 'Скучно', 'Мусор', 'Несанкционированная реклама', 'Несанкционированная реклама', 'Несанкционированная реклама', 'Шум', 'Мусор', 'Мусор', 'Скучно', 'Шум', 'Несанкционированная реклама', 'Шум', 'Несанкционированная реклама', 'Мусор', 'Мусор', 'Скучно', 'Безопасность', 'Мусор', 'Несанкционированная реклама', 'Мусор', 'Скучно', 'Мусор', 'Скучно', 'Безопасность', 'Несанкционированная реклама', 'Мусор', 'Мусор', 'Несанкционированная реклама', 'Несанкционированная реклама', 'Скучно', 'Скучно', 'Мусор', 'Безопасность', 'Скучно', 'Скучно', 'Безопасность', 'Безопасность', 'Безопасность', 'Шум', 'Мусор', 'Скучно', 'Несанкционированная реклама', 'Несанкционированная реклама', 'Мусор', 'Шум', 'Скучно', 'Мусор', 'Скучно', 'Мусор', 'Скучно', 'Мусор', 'Несанкционированная реклама', 'Шум', 'Шум', 'Скучно', 'Несанкционированная реклама', 'Скучно', 'Мусор', 'Шум', 'Мусор', 'Скучно', 'Скучно', 'Скучно', 'Мусор', 'Скучно', 'Несанкционированная реклама', 'Мусор', 'Скучно', 'Мусор', 'Шум', 'Несанкционированная реклама', 'Мусор', 'Мусор', 'Шум', 'Шум', 'Скучно', 'Шум', 'Мусор', 'Скучно', 'Транспорт', 'Безопасность', 'Несанкционированная реклама', 'Безопасность', 'Безопасность', 'Скучно', 'Несанкционированная реклама', 'Скучно', 'Мусор',

'Несанкционированная реклама', 'Скучно', 'Несанкционированная реклама', 'Мусор', 'Скучно', 'Мусор', 'Скучно', 'Скучно', 'Несанкционированная реклама', 'Скучно', 'Безопасность', 'Мусор', 'Мусор', 'Несанкционированная реклама', 'Скучно', 'Скучно', 'Несанкционированная реклама', 'Мусор', 'Скучно', 'Скучно', 'Мусор', 'Мусор', 'Шум', 'Транспорт', 'Несанкционированная реклама', 'Несанкционированная реклама', 'Шум', 'Безопасность', 'Несанкционированная реклама', 'Мусор', 'Скучно', 'Мусор', 'Скучно', 'Скучно', 'Скучно', 'Скучно', 'Мусор', 'Несанкционированная реклама', 'Мусор', 'Несанкционированная реклама', 'Шум', 'Безопасность', 'Шум', 'Несанкционированная реклама', 'Шум', 'Шум', 'Безопасность', 'Несанкционированная реклама', 'Шум', 'Мусор', 'Шум', 'Шум', 'Скучно', 'Несанкционированная реклама', 'Скучно', 'Мусор', 'Скучно', 'Шум', 'Несанкционированная реклама', 'Мусор']

### Решение

Возможное решение в интерактивной тетради - построить график.

```
import matplotlib.pyplot as plt
data = данныеизусловия
plt.bar([x for x in set(data)], [data.count(x) for x in set(data)])
plt.gcf().autofmt_xdate()
```

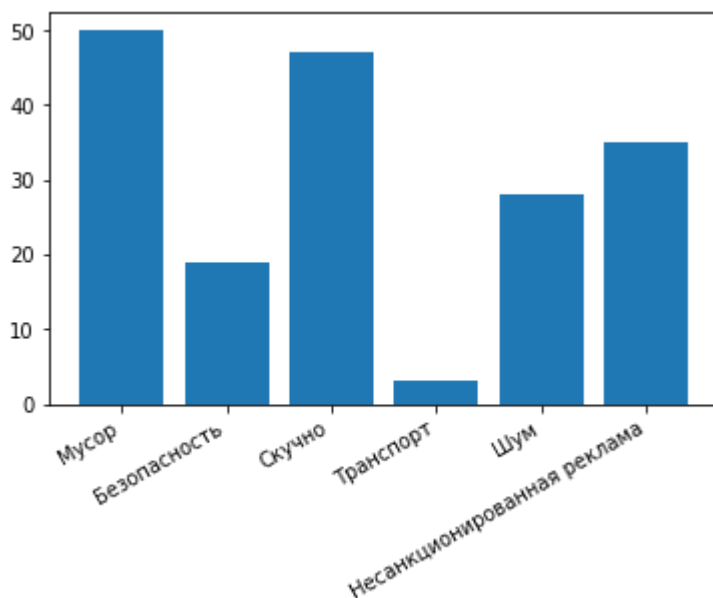


Рис. 38. График выраженности проблем территории

Второе возможное решение - построить облако слов.

```
text = данныеизусловия
```

```
stopwords = set(STOPWORDS)
stopwords.update(["на", "все", "эта", "не", "уже", "это", "что", "по", "так", "но"])
wordcloud = WordCloud(stopwords = stopwords,
background_color="white").generate(text)
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.show()
```

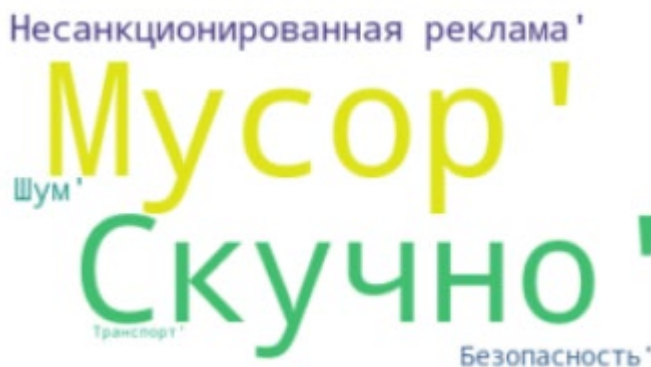


Рис. 38. Облако слов по проблемам территории

Ответ: Мусор и Скучно

## Задача 5. Формирование рейтинга мест

### Условие

Рейтинг выстраивается по количеству лайков, жалоб, по посещаемости места.

Если количество лайков и показатель посещаемости места увеличивает его рейтинг, то количество жалоб уменьшает. Например, если у места 20 лайков, 40 посетителей и 10 жалоб, то итоговый рейтинг будет равен 50.

Воспользуйтесь графиком показателей мест (рис. 39) или данными по каждому показателю (likes — лайки, problems — проблемы, attendance — посещаемость, labels — названия мест), представленными ниже, и сформируйте рейтинг.

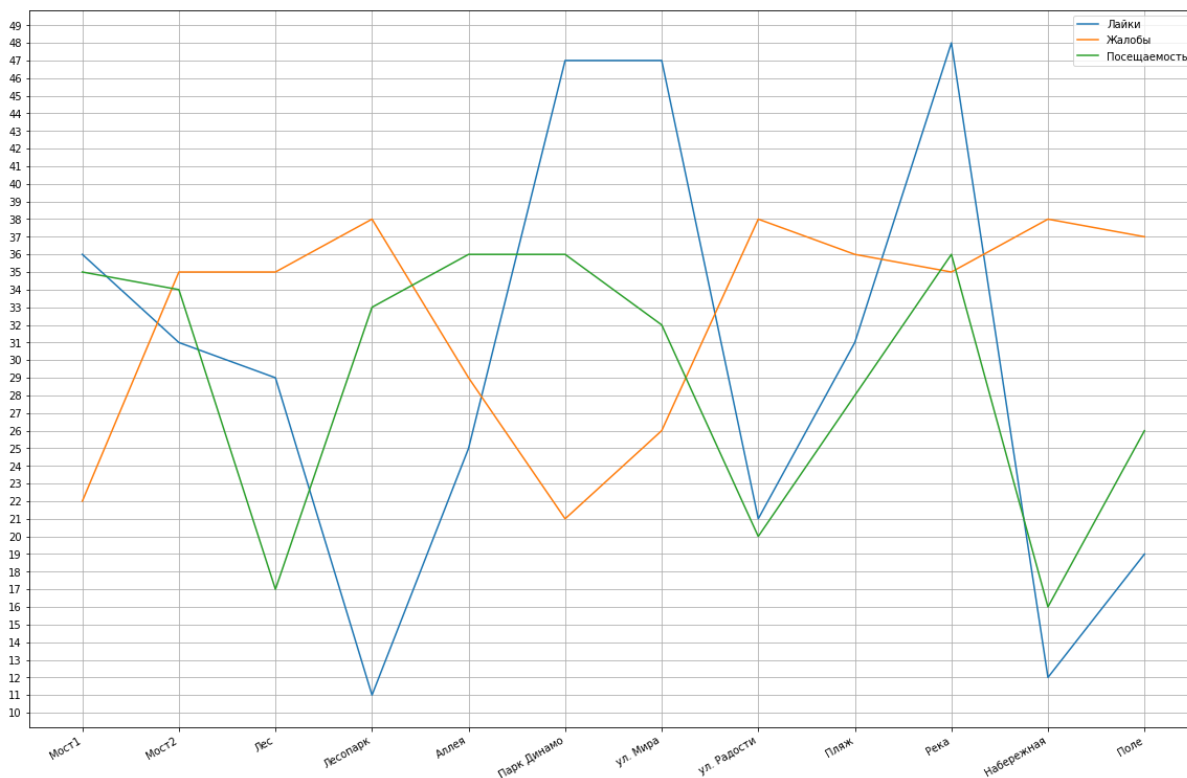


Рис. 39. График количества отзывов и их характер

В ответе укажите топовое и худшее место, по версии пользователей.

Примечание. При вводе значений регистр не учитывается.

### Решение.

Для определения лучшего места нам необходимо посмотреть максимальное значение лайков(синяя линия) и посещаемости(зеленая линия) и при этом меньшее количество жалоб(оранжевая линия).

На графике видно, что максимальное количество лайков приходится на места: парк Динамо, ул. Мира и Река. При этом у ул. Мира меньшее количество посещаемости и больше жалоб по сравнению с Парком Динамо, поэтому вычеркиваем ее из кандидатов в самое “топовое” место. Осталось сравнить количество жалоб на Парк Динамо и Реку. По графику видно, что в парке их гораздо меньше, но можно посчитать по формуле из описания задания.

$$\text{Река: } 48 + 36 - 35 = 49$$

Парк Динамо:  $47+36-21=62$

Таким образом самое “топовое” место - Парк Динамо

Для определения худшего места нужно смотреть наименьшие значения синей и зеленой линий (лайков и посещаемости) и одновременно с этим большое высокое значение оранжевой (жалоб). Из графика видно, что Набережная имеет как раз такие значения. И ближайшее место по значению лайков к ней Лесопарк, имеет большую посещаемость, что убирает его из конкурентов на звание “худшего” места.

Вывод: Худшее место - Набережная

#### Ответ

Худшее место: Набережная

Лучшее место: Парк Динамо

#### Задача 6. Что важнее?

##### Условие.

В задаче № 4 стажерами были выделены наиболее острые проблемы района. Администрация района предложила определить жителям приоритет финансового вложения в решение этих проблем.

Можно выделить одну проблему и потратить выделенные финансы только на одно решение, либо выбрать две с распределением финансов.

Часть жителей проголосовали только за финансирование решений по проблеме «А», еще часть только за решение по проблеме «Б», но были те, кто считает важным, одновременное решение обеих проблем. Таким образом, 75% всех жителей высказались за решение проблемы «Б» на территории, 85% — за устранение проблемы «А».

В ответе укажите, сколько процентов жителей района предложили решать обе проблемы одновременно?

## Решение

По условию общее значение проголосовавших за:

“А” = 85%

“Б” = 75%

Всего 100% проголосовавших, соответственно, все, кто не проголосовал за “Б” проголосовали за “А”. Вычисляем это значение:  $100 - 75 = 25$ .

25% — это точно проголосовавшие только за вариант “А”, поэтому нужно вычесть это значение из общего значения проголосовавших за “А”.  $85 - 25 = 60\%$ . 60% - и есть искомое значение.

Можно проверить его: есть 60 процентов проголосовали за оба варианта, значит за вариант “А” проголосовали  $85 - 60 = 25$ , а за “Б”  $75 - 60 = 15$ .

$15 + 25 + 60 = 100\%$

**Ответ:** 60%

## Задача 7. Умные решения

### Условие.

Для того, чтобы территория процветала, и среда обитания населения города была комфортной, а люди счастливыми, необходимо внедрить несколько умных решений.

Каждое умное решение имеет оценку, указывающую на степень одобрения его жителями района — баллы счастья (см. условие задачи 9 первой попытки).

Администрация может:

- разместить пункты сбора раздельного мусора (waste) — 1000 баллов за один пункт;
- открыть общепиты/фудкорты (safe) — 3000 баллов за одно предприятие;
- настроить вай-фай на площадках (wifi) — 1500 баллов за одну площадку;

- установить кнопки экстренной помощи (helps) — 600 баллов за один двор.

Кроме того, умные решения требуют финансовых затрат, которые у нас ограничены. Городом администрации района выделено — 170 000 условных денежных единиц (уде), при этом:

- на пункт мусора тратится 7000 уде;
- на предприятие общественного питания — 5000 уде;
- на wi-fi точку — 30000 уде;
- на кнопку помощи — 10000 уде.

Определите оптимальное количество умных решений достаточных для обеспечения комфортной среды обитания района и укладывающихся в бюджет.

Известно, что администрация намерена реализовать все умные решения, а это значит, что количество размещаемых объектов (кафе, точки wi-fi, пункты сбора раздельного мусора, кнопки экстренной помощи) не могут быть равны нулю.

Параметр необходимого уровня комфорта устанавливается таким образом, чтобы его значение было не ниже максимального показателя в других районах города (жители которых довольны обустройством и жизнеобеспечением территории): 39020, 40000, 38800, 39800.

Примечание. В ответе перечислите через пробел — соответствующие значения для waste, cafe, wifi и helps.

## Решение

Решение может быть разное, главное - чтобы подходило под условия задачи. А именно:

- общая сумма не превышает 170 000 уде;
- количество мусорок, общепитов, вай-фая и кнопки больше 0;

- уровень комфорта больше 40000.

При стоимости и комфорту за единицу:

- пункт сбора мусора - 7000 уде и 1000 баллов;
- общепит - 5000 уде и 3000 баллов;
- вай-фай - 30000 уде и 1500 баллов;
- кнопка помощи 10000 уде и 600 баллов.

*Вариант ответа:*

Первым действием посчитаем стоимость и баллы, когда каждого решения по 1, так как это обязательное условие.

уде:  $7000+5000+30000+10000 = 52000$

баллы:  $1000+3000+1500+600 = 6100$

Посчитаем оставшуюся сумму для реализации решений и недостающие баллы.

остаток баланса:  $170000-52000 = 118000$

не хватает баллов:  $40000-6100 = 33900$

И теперь нужно определить, что нужно реализовать на сумму 118 000 уде для получения 6100 баллов.

Например, мы поставим еще 4 пункта сбора мусора.

остаток баланса:  $118000-(7000*4) = 90000$

не хватает баллов:  $33900-(1000*4) = 29900$

И допустим сделаем еще одну точку вай-фая.

остаток баланса:  $90000-3000 = 87000$

не хватает баллов:  $29900-1500 = 28400$

А на оставшиеся уде реализуем общепиты(17 штук)

остаток баланса:  $87000-(5000*17) = 2000$

не хватает баллов:  $28400-(3000*17) = -22600$



Так как значение баллов ушло в отрицательную отметку, значит получившееся решение превышает максимальное количество баллов комфорта в других регионах, чего и нужно было добиться. Сумму на реализацию проектов мы не превысили, и каждого решение минимум по 1.

Таким образом одним из возможных вариантов ответа является: 5 точек сбора мусора, 18 общепитов, 2 точки вай-фая и 1 кнопка экстренной помощи.

*Возможно решение в интерактивной тетради с помощью ползунков*

```
from ipywidgets import Dropdown, Checkbox, HBox, VBox, Text, Button
from IPython.display import display, clear_output
```

```
def counts_of_points(waste = 10, cafe = 10, wifi = 10, helps = 10):
```

```
    mwaste = waste*7000
```

```
    mcafe = cafe*5000
```

```
    mwifi = wifi*30000
```

```
    mhelp = helps*10000
```

```
    spends = [mwaste, mcafe, mwifi, mhelp]
```

```
    maxspend = 170000
```

```
    title_1 = "Текущие траты %d!" % sum(spends) if sum(spends) < maxspend else
"Текущие траты %d! Превышают лимит 170-000" % sum(spends)
```

```
    max_happy = waste*1000 + wifi*1500 + cafe*3000 + helps*600
```

```
    other_centers = [39020, 40000, 38800, 39800, max_happy ]
```

```
    explode = explode = (0, 0, 0, 0, 0.1)
```

```
    title_2 = "Счастья слишком мало!" if max(other_centers[:-1]) > other_centers[-1]
else "Счастья сколько надо!"
```

```
    plt.figure(figsize = (10, 4))
```

```
    plt.subplot(1,2,1)
```

```
    plt.pie(spends, labels=spends, autopct='%1.1f%%',
        shadow=True, startangle=90)
```

```
    plt.title(title_1)
```

```
    plt.subplot(1,2,2)
```

```
plt.pie(other_centers, labels=other_centers, explode = explode,  
autopct='%1.1f%%', shadow=True, startangle=90)  
plt.title(title_2)  
plt.tight_layout()  
plt.show()
```

```
import matplotlib.pyplot as plt  
from ipywidgets import interact, interactive, fixed, interact_manual  
from IPython.display import display
```

```
w = interactive(counts_of_points, waste = 5, cafe = 5, wifi = 5, helps = 5)  
display(w)
```

**Ответ:** возможный вариант ответа 5 18 2 1

## **Задача 8. Плотность предприятий в районе**

### **Условие.**

Исходя из выявленных проблем и определенных умных решений, нужно порекомендовать администрации, в каком районе лучше открыть заведение общественного питания.

На картинке (рис. 40) представлено 6 общепитов в 3 районах (прямоугольники разного цвета).

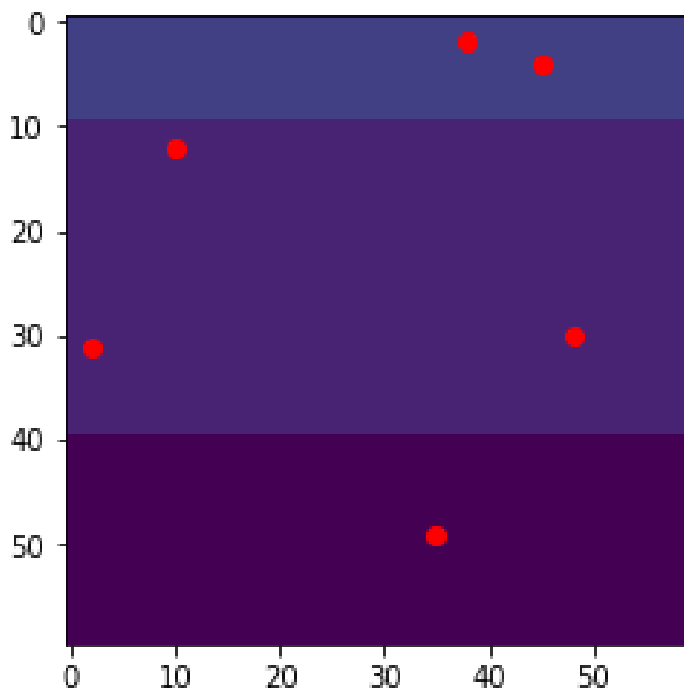


Рис. 40. Распределение пунктов общественного питания в районах.

Нужно определить плотность распределения общепитов в районах. Для этого напишите функцию, на вход которой подается 3 параметра: а (высота прямоугольника), b (ширина прямоугольника), count (число кафе в районе).

Входные данные по 3 районам выглядят так:  $[[10, 60, 2], [30, 60, 3], [20, 60, 1]]$ .

На выходе функция должна подсчитывать значение плотности, которое вычисляется как число заведений на площадь района:

$S = a \times b$ , где S – площадь.

$D = \text{count} / S$ , где D - плотность заведений.

Найдите D для каждого района, введите значения через запятую для первого, второго и третьего района, оставив после нуля 4 знака, остальное отбросив, например: 0.0039666666666666668 — получится 0.0039 (между нулем и дробной частью – точка)

Пример ответа: 0.0259,0.0007,0.0340

**Решение**

По условию на вход поступают следующие данные:  $[[10, 60, 2], [30, 60, 3], [20, 60, 1]]$ . Распишем их, более подробно.

Район №1:

$$a = 10$$

$$b = 60$$

$$\text{count} = 2$$

Район №2:

$$a = 30$$

$$b = 60$$

$$\text{count} = 3$$

Район №3:

$$a = 20$$

$$b = 60$$

$$\text{count} = 1$$

Теперь по формулам нужно вычислить площадь и плотность по каждому из районов.

Район №1:

$$S = 10 * 60 = 600$$

$$D = 2/600 = 0.003333333333, \text{ оставляем 4 знака после запятой путем отбрасывания } D = 0.0033$$

Район №2:

$$S = 30 * 60 = 1800$$

$$D = 3/1800 = 0.001666666666, \text{ оставляем 4 знака после запятой путем отбрасывания } D = 0.0016$$

Район №3:

$$S = 20 * 60 = 1200$$

$D = 1/1200 = 0.000833333333$ , оставляем 4 знака после запятой путем отбрасывания  $D = 0.0008$

Ответ: 0.0033, 0.0016, 0.0008

### Задача 9. Где открыть ресторан?

#### Условие.

В задаче 8 junior-стажеры определили какова плотность предприятий общественного питания в трех районах города Энск. Теперь научный руководитель просит проделать их дополнительную работу и найти район, в котором открытие нового предприятия общественного питания, например ресторана, будет наиболее целесообразным.

По каждому району представлена посещаемость, ее значение указано для каждой клетки территории (рис. 41). Красные точки на графике — это уже открытые заведения общественного питания.

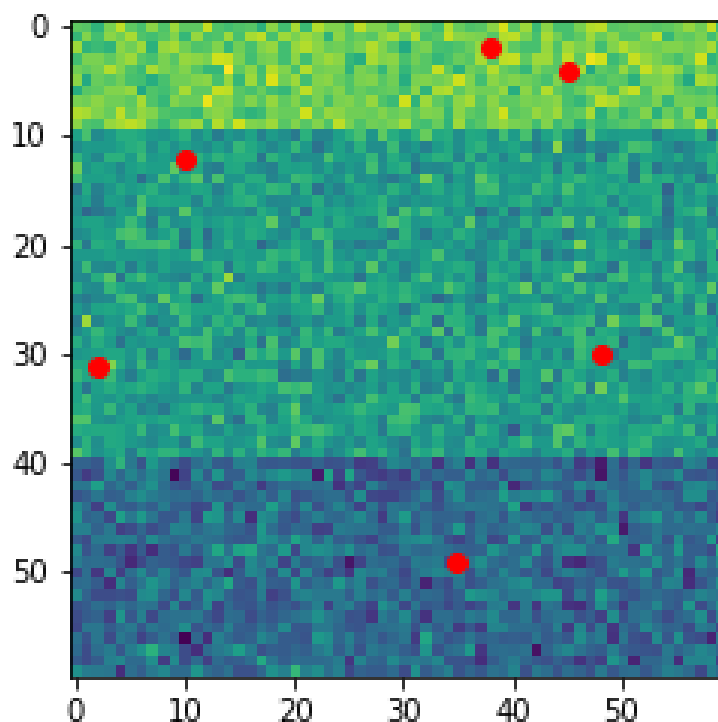


Рис. 41. Посещаемость районов и расположение заведений общественного питания.

В первую очередь, стажерам необходимо посчитать среднюю посещаемость по району, т. е. следует найти среднее значение по всем точкам прямоугольника.

Значения посещаемости по районам содержатся в переменных с названиями:

для первого района (район №1) — data\_1;

для второго (район №2) — data\_2;

для третьего (район №3) — data\_3.

Посещаемость по району — это спрос, который может быть удовлетворен предложением — открытыми заведениями общепита.

Ваша задача найти показатель рейтинга района, который показывает, насколько уместно открывать в районе новое заведение общественного питания.

Чем больше средняя посещаемость по району, тем выше этот показатель, а, значит, в этом районе спрос на открытие новых предприятий больше. С другой стороны, чем больше плотность количества заведений в районе, тем показатель ниже. Поэтому рейтинг районов нужно подсчитать следующим образом:

$R=A/D$ , где  $A$  — среднее значение посещаемости по району,  $D$  — плотность заведений района (из задачи №8 с уже отброшенной дробной частью).

Район с максимальным значением  $R$  — самый пригодный для открытия в нем нового заведения общественного питания.

Определите рейтинг районов. В ответе укажите район, который находится в топе и после запятой значение рейтинга для него (дробную часть при подсчете рейтинга просто отбросьте).

*Пример ответа: 1,43355, где 1 — это район, а 43355 — значение рейтинга.*

### Данные.

```
data_1 = [58, 57, 51, 61, 68, 58, 55, 63, 56, 65, 58, 58, 69, 56, 54, 56, 57, 50, 63, 54, 66, 53, 67, 62, 62, 57, 68, 62, 62, 57, 65, 61, 58, 54, 67, 59, 56, 57, 63, 71, 58, 65, 53, 63, 55, 55, 61, 67, 68, 57, 58, 59, 58, 57, 58, 60, 64, 60, 69, 53, 52, 62, 54, 63, 60, 66, 56, 54, 57, 62, 52, 64, 55, 56, 65, 55, 66, 68, 55, 65, 61, 65, 62, 51, 63, 57, 59, 57, 63, 66, 59, 67, 55, 57, 69, 63, 62, 61, 57, 67, 68, 61, 55, 66, 60, 55, 58, 65, 59, 63, 57, 57, 62, 68, 52, 59, 58, 61, 54, 60, 65, 51, 53, 51, 63, 60, 65, 50, 64, 66, 60, 56, 57, 62, 52, 60, 55, 62, 64, 54, 64, 67, 63, 59, 60, 58, 61, 67, 62, 57, 48, 59, 61, 55, 54, 56, 59, 64, 65, 65, 58, 55, 55, 64, 56, 65, 61, 63, 54, 62, 67, 63, 61, 60, 62, 62, 60, 62, 66, 54, 54, 61, 61, 58, 62, 65, 57, 55, 61, 57, 64, 63, 61, 70, 69, 61, 54, 60, 61, 54, 63, 54, 62, 65, 63, 52, 49, 66, 62, 63, 55, 60, 57, 52, 55, 59, 57, 66, 57, 68, 69, 58, 65, 65, 57, 68, 58, 73, 70, 53, 57, 69, 64, 61, 58, 55, 59, 70, 51, 55, 60, 58, 60, 67, 62, 56, 53, 66, 61, 59, 62, 57, 52, 52, 75, 58, 63, 54, 67, 62, 55, 67, 60, 61, 61, 51, 61, 59, 52, 61, 52, 58, 57, 69, 58, 50, 60, 58, 67, 59, 57, 52, 62, 58, 57, 61, 66, 48, 58, 56, 64, 57, 69, 63, 66, 63, 55, 61, 56, 55, 63, 48, 59, 57, 60, 55, 54, 49, 55, 60, 60, 62, 63, 55, 64, 52, 65, 55, 71, 60, 65, 54, 62, 63, 59, 56, 61, 62, 57, 60, 49, 61, 62, 60, 62, 70, 62, 54, 63, 60, 62, 60, 58, 55, 62, 55, 56, 57, 55, 58, 57, 50, 61, 61, 63, 56, 48, 66, 64, 58, 49, 57, 58, 62, 62, 64, 47, 56, 54, 58, 66, 58, 58, 69, 68, 57, 55, 59, 54, 52, 60, 68, 61, 59, 62, 56, 59, 57, 58, 59, 54, 63, 51, 66, 67, 58, 62, 60, 60, 64, 49, 63, 57, 67, 58, 55, 59, 64, 63, 55, 65, 69, 56, 63, 64, 59, 57, 55, 57, 48, 61, 60, 63, 60, 60, 58, 66, 56, 52, 62, 59, 59, 73, 62, 63, 57, 62, 57, 64, 58, 62, 56, 61, 55, 61, 62, 59, 64, 56, 53, 70, 51, 65, 58, 54, 63, 66, 57, 63, 58, 58, 61, 60, 61, 63, 55, 52, 58, 68, 66, 59, 58, 57, 61, 52, 60, 60, 59, 60, 54, 67, 66, 62, 59, 68, 59, 64, 62, 56, 54, 54, 52, 61, 63, 61, 54, 51, 53, 61, 59, 59, 70, 58, 64, 63, 60, 60, 59, 62, 59, 61, 57, 62, 61, 52, 63, 58, 56, 62, 70, 57, 59, 64, 61, 65, 56, 58, 66, 50, 58, 52, 56, 62, 50, 59, 55, 63, 51, 64, 61, 63, 56, 55, 64, 66, 68, 66, 57, 52, 58, 63, 65, 52, 67, 59, 70, 62, 61, 65, 57, 62, 57, 62, 56, 68, 61, 56, 60, 59, 61, 66, 56, 54, 59, 51, 65, 54, 57, 62, 64, 61, 60, 54, 62, 53, 57, 50, 56, 52, 55, 57, 58, 54, 60, 64, 57, 58, 67, 57, 69]
```

```
data_2 = [46, 55, 39, 43, 49, 49, 47, 43, 42, 46, 50, 40, 46, 35, 43, 40, 46, 36, 44, 40, 34, 50, 37, 37, 45, 50, 55, 42, 58, 49, 44, 50, 59, 48, 49, 40, 48, 47, 41, 54, 44, 55, 37, 46, 52, 43, 36, 58, 46, 44, 45, 45, 45, 35, 43, 45, 45, 51, 32, 42, 40, 44, 43, 43, 46, 49, 51, 40, 47, 41, 46, 42, 39, 45, 40, 52, 48, 43, 38, 43, 52, 43, 32, 40, 46, 40, 50, 40, 40, 53, 39, 43, 36, 44, 48, 40, 48, 51, 42, 44, 47, 42, 35, 40, 63, 39, 34, 41, 36, 37, 48, 55, 53, 45, 40, 34, 38, 44, 45, 53, 51, 44, 38, 50, 50, 44, 53, 43, 45, 42, 45, 50, 45, 51, 47, 49, 49, 33, 51, 46, 37, 40, 40, 55, 46, 44, 44, 52, 39, 49, 44, 37, 46, 50, 49, 37, 51, 39,
```

47, 37, 45, 54, 42, 44, 39, 43, 44, 52, 32, 48, 53, 39, 46, 45, 42, 48, 39, 46, 50, 49, 60,  
42, 46, 44, 38, 45, 47, 45, 45, 58, 41, 48, 37, 35, 47, 48, 41, 39, 36, 46, 45, 46, 40, 46,  
39, 44, 41, 54, 40, 47, 50, 33, 39, 38, 40, 49, 41, 50, 45, 43, 43, 42, 43, 48, 34, 48, 37,  
46, 44, 51, 51, 34, 49, 33, 41, 37, 39, 37, 50, 47, 54, 45, 45, 46, 40, 44, 31, 41, 55, 44,  
51, 57, 37, 55, 42, 42, 49, 49, 37, 42, 53, 51, 47, 39, 42, 50, 40, 42, 37, 34, 40, 44, 61,  
49, 35, 44, 41, 58, 50, 41, 47, 45, 43, 51, 45, 48, 52, 48, 46, 43, 47, 34, 53, 47, 49, 41,  
47, 33, 39, 55, 34, 48, 36, 42, 48, 44, 46, 50, 43, 39, 44, 47, 46, 36, 42, 50, 44, 46, 52,  
50, 38, 49, 32, 42, 36, 44, 47, 43, 40, 50, 45, 38, 46, 49, 42, 46, 41, 53, 54, 42, 44, 42,  
53, 48, 44, 55, 45, 47, 45, 49, 44, 47, 48, 48, 28, 46, 40, 49, 45, 47, 41, 45, 35, 45, 44,  
38, 55, 46, 47, 46, 46, 37, 47, 40, 45, 49, 51, 44, 38, 39, 41, 43, 49, 39, 43, 39, 44, 46,  
51, 42, 39, 43, 41, 44, 43, 48, 40, 47, 45, 42, 48, 51, 42, 45, 38, 48, 45, 47, 49, 37, 47,  
43, 45, 50, 33, 47, 42, 46, 37, 39, 40, 29, 33, 36, 45, 44, 57, 37, 41, 39, 57, 44, 39, 41,  
40, 41, 37, 49, 45, 38, 50, 32, 47, 45, 47, 44, 46, 47, 52, 37, 55, 47, 55, 49, 39, 44, 43,  
30, 38, 48, 32, 43, 40, 32, 49, 42, 42, 41, 45, 46, 36, 46, 45, 49, 31, 46, 49, 53, 33, 44,  
43, 46, 46, 42, 45, 53, 45, 35, 50, 40, 42, 47, 42, 45, 48, 45, 48, 47, 58, 40, 36, 47, 47,  
40, 49, 46, 41, 38, 52, 56, 50, 42, 43, 47, 47, 41, 58, 39, 46, 38, 40, 51, 42, 50, 51, 58,  
43, 56, 41, 47, 52, 58, 49, 42, 33, 44, 40, 45, 37, 36, 39, 41, 46, 45, 54, 56, 53, 40, 45,  
51, 34, 48, 30, 42, 47, 44, 49, 47, 42, 38, 46, 40, 54, 50, 44, 44, 44, 44, 41, 35, 51, 42,  
33, 43, 41, 50, 46, 41, 36, 39, 46, 38, 47, 46, 43, 33, 41, 43, 46, 39, 41, 47, 34, 39, 36,  
53, 40, 43, 52, 44, 51, 34, 49, 47, 40, 52, 53, 53, 55, 49, 32, 47, 34, 42, 51, 44, 45, 29,  
37, 44, 49, 45, 40, 54, 38, 40, 34, 37, 36, 51, 49, 44, 45, 42, 46, 41, 46, 37, 44, 50, 43,  
35, 45, 38, 43, 47, 62, 44, 47, 43, 52, 39, 28, 46, 42, 59, 50, 43, 47, 53, 43, 43, 43, 46,  
44, 44, 43, 42, 46, 44, 45, 55, 42, 43, 42, 46, 44, 44, 52, 49, 52, 45, 42, 53, 43, 40, 45,  
42, 50, 41, 43, 41, 46, 55, 42, 43, 36, 38, 43, 49, 39, 47, 50, 33, 41, 35, 43, 41, 49, 45,  
44, 43, 40, 47, 41, 49, 51, 42, 49, 30, 40, 53, 39, 38, 47, 35, 45, 39, 46, 41, 41, 42, 47,  
51, 34, 48, 45, 53, 51, 41, 47, 51, 42, 55, 46, 48, 40, 57, 40, 46, 36, 41, 48, 44, 46, 49,  
47, 35, 48, 43, 47, 39, 49, 45, 48, 46, 59, 46, 44, 32, 53, 44, 46, 36, 46, 57, 38, 33, 49,  
49, 39, 38, 49, 57, 32, 49, 54, 46, 40, 38, 36, 49, 37, 48, 65, 35, 44, 43, 44, 43, 42, 52,  
47, 44, 42, 43, 32, 45, 48, 43, 37, 49, 55, 42, 48, 41, 47, 32, 51, 49, 44, 34, 45, 41, 49,  
58, 36, 38, 34, 52, 55, 46, 51, 39, 50, 45, 47, 37, 43, 46, 48, 40, 39, 52, 47, 42, 42, 45,  
47, 43, 53, 44, 44, 58, 42, 45, 42, 33, 39, 34, 46, 39, 36, 48, 51, 51, 57, 48, 49, 46, 51,  
46, 44, 46, 56, 50, 43, 38, 46, 44, 51, 59, 38, 51, 46, 44, 55, 45, 49, 44, 40, 47, 47, 37,  
41, 54, 48, 43, 58, 36, 37, 34, 38, 39, 53, 55, 41, 37, 58, 46, 48, 31, 42, 46, 52, 54, 52,  
37, 46, 44, 51, 46, 59, 40, 48, 52, 45, 44, 50, 47, 52, 48, 41, 29, 35, 43, 38, 33, 54, 45,  
41, 44, 45, 37, 41, 48, 42, 41, 56, 40, 48, 48, 48, 54, 37, 50, 39, 49, 39, 40, 43, 38, 46,  
47, 45, 48, 54, 45, 40, 54, 50, 46, 46, 45, 50, 40, 49, 54, 42, 41, 46, 47, 52, 38, 44, 49,



41, 49, 46, 43, 33, 41, 50, 51, 45, 50, 39, 44, 46, 39, 48, 47, 50, 54, 39, 38, 43, 47, 39,  
43, 39, 37, 53, 35, 45, 45, 36, 54, 31, 44, 35, 65, 37, 43, 42, 53, 43, 34, 41, 40, 53, 44,  
39, 51, 46, 51, 37, 40, 40, 33, 41, 44, 37, 44, 44, 47, 34, 48, 36, 44, 41, 53, 50, 30, 51,  
50, 46, 34, 49, 47, 57, 40, 44, 47, 50, 48, 51, 53, 36, 60, 48, 47, 57, 55, 34, 44, 44, 47,  
45, 35, 39, 44, 52, 46, 40, 37, 45, 48, 36, 49, 44, 41, 54, 45, 51, 39, 51, 42, 46, 43, 36,  
52, 43, 52, 35, 41, 46, 55, 46, 42, 52, 43, 46, 57, 32, 53, 45, 34, 55, 54, 43, 37, 54, 40,  
34, 51, 54, 34, 49, 47, 39, 40, 32, 44, 56, 44, 39, 50, 43, 43, 49, 41, 49, 44, 32, 43, 41,  
44, 43, 42, 55, 53, 54, 41, 38, 39, 48, 32, 55, 36, 42, 51, 45, 48, 48, 45, 41, 42, 42, 60,  
46, 49, 32, 42, 49, 41, 51, 48, 54, 44, 47, 34, 43, 43, 43, 43, 54, 49, 48, 51, 31, 44, 41,  
39, 43, 59, 52, 38, 44, 48, 42, 44, 45, 47, 47, 42, 50, 36, 49, 40, 30, 43, 53, 39, 43, 27,  
53, 47, 53, 43, 57, 37, 55, 44, 37, 48, 43, 42, 44, 49, 43, 49, 51, 33, 50, 51, 52, 45, 42,  
51, 56, 39, 47, 38, 47, 36, 53, 48, 50, 34, 44, 44, 41, 52, 52, 50, 51, 41, 51, 40, 48, 47,  
57, 62, 49, 40, 53, 41, 46, 44, 40, 44, 44, 43, 41, 49, 43, 44, 43, 34, 47, 37, 55, 40, 37,  
44, 52, 59, 38, 50, 46, 34, 41, 47, 37, 50, 42, 47, 53, 46, 51, 46, 43, 44, 46, 43, 50, 46,  
45, 42, 37, 53, 52, 46, 43, 38, 44, 47, 43, 40, 52, 47, 46, 53, 42, 35, 45, 49, 47, 44, 49,  
49, 37, 39, 55, 50, 50, 53, 52, 31, 35, 34, 42, 41, 34, 48, 33, 46, 40, 41, 39, 45, 38, 42,  
35, 48, 43, 45, 31, 52, 31, 45, 43, 56, 34, 51, 46, 40, 44, 49, 36, 42, 38, 46, 42, 45, 44,  
49, 52, 36, 46, 33, 45, 49, 45, 47, 50, 47, 43, 41, 51, 37, 46, 46, 34, 41, 43, 47, 45, 54,  
52, 55, 44, 40, 42, 37, 41, 45, 37, 46, 39, 50, 35, 38, 52, 43, 51, 47, 42, 43, 34, 46, 44,  
33, 43, 48, 55, 34, 60, 44, 50, 42, 41, 45, 50, 40, 51, 57, 45, 52, 46, 53, 44, 42, 46, 39,  
42, 37, 41, 40, 42, 42, 42, 37, 41, 47, 45, 46, 40, 41, 49, 38, 45, 39, 32, 45, 40, 54, 47,  
46, 47, 35, 48, 50, 39, 34, 53, 50, 46, 42, 50, 44, 41, 50, 41, 49, 42, 47, 44, 46, 56, 46,  
45, 48, 44, 45, 50, 49, 49, 54, 48, 49, 43, 52, 41, 46, 36, 40, 34, 44, 47, 51, 47, 48, 40,  
44, 50, 38, 36, 48, 58, 40, 43, 42, 45, 47, 52, 47, 46, 43, 55, 54, 47, 46, 49, 41, 51, 50,  
52, 38, 42, 48, 48, 34, 47, 41, 39, 45, 45, 44, 47, 40, 45, 46, 45, 34, 51, 37, 43, 41, 56,  
56, 42, 40, 44, 46, 47, 45, 48, 46, 45, 42, 35, 57, 38, 31, 50, 51, 37, 46, 54, 47, 38, 42,  
43, 43, 47, 44, 35, 53, 54, 50, 38, 36, 41, 50, 35, 45, 41, 44, 42, 42, 41, 49, 45, 40, 48,  
47, 49, 48, 43, 37, 41, 39, 53, 45, 42, 42, 47, 42, 49, 47, 48, 48, 47, 44, 52, 39, 41, 54,  
46, 48, 39, 38, 46, 40, 41, 49, 46, 44, 51, 52, 54, 51, 46, 40, 47, 45, 34, 49, 50, 43, 41,  
50, 42, 42, 34, 51, 46, 37, 25, 44, 37, 39, 56, 58, 49, 35, 47, 45, 41, 48, 47, 50, 48, 45,  
53, 37, 50, 36, 40, 50, 40, 48, 47, 43, 41, 38, 43, 44, 35, 53, 46, 46, 41, 42, 45, 56, 40,  
38, 58, 39, 49, 42, 39, 44, 37, 46, 43, 38, 34, 47, 41, 47, 40, 39, 50, 44, 45, 38, 47, 53,  
43, 43, 56, 40, 51, 45, 48, 45, 44, 49, 52, 42, 41, 42, 44, 48, 34, 34, 57, 50, 41, 45, 48,  
35, 49, 46, 51, 51, 45, 42, 37, 51, 40, 50, 40, 51, 47, 52, 35, 43, 46, 35, 40, 47, 54, 46,  
50, 41, 32, 51, 47, 52, 39, 41, 49, 48, 37, 47, 45, 42, 36, 44, 44, 49, 33, 44, 40, 54, 41,  
53, 48, 41, 39, 54, 43, 38, 44, 46]

data\_3 = [28, 28, 25, 40, 51, 34, 24, 43, 35, 29, 22, 31, 36, 26, 25, 36, 37, 26, 25, 26,  
24, 23, 28, 23, 24, 28, 18, 19, 28, 38, 35, 29, 33, 17, 31, 36, 16, 34, 16, 30, 29, 28, 28,  
30, 24, 33, 18, 22, 11, 31, 20, 29, 17, 27, 25, 24, 37, 34, 20, 20, 25, 18, 23, 38, 29, 37,  
31, 38, 33, 7, 20, 39, 26, 42, 27, 27, 39, 33, 39, 43, 38, 38, 11, 46, 20, 46, 25, 20, 18,  
22, 18, 34, 25, 30, 27, 27, 38, 29, 32, 22, 32, 34, 29, 38, 35, 26, 22, 29, 10, 30, 25, 32,  
38, 31, 32, 40, 25, 24, 27, 23, 33, 33, 45, 29, 34, 27, 35, 41, 31, 25, 23, 31, 35, 22, 23,  
35, 30, 18, 27, 31, 30, 32, 31, 33, 16, 26, 20, 36, 28, 23, 23, 25, 30, 20, 41, 38, 27, 27,  
26, 41, 14, 23, 28, 30, 27, 28, 39, 32, 36, 30, 28, 30, 39, 26, 22, 26, 37, 39, 34, 30, 24,  
27, 33, 27, 28, 36, 40, 35, 32, 30, 31, 28, 34, 30, 31, 24, 29, 35, 27, 38, 23, 25, 29, 31,  
31, 23, 40, 21, 20, 20, 25, 30, 36, 22, 32, 40, 28, 31, 32, 45, 35, 38, 37, 44, 34, 24, 32,  
18, 31, 36, 28, 29, 29, 32, 29, 26, 40, 40, 25, 27, 39, 31, 19, 40, 28, 26, 31, 27, 35, 27,  
27, 21, 22, 42, 23, 33, 30, 30, 45, 33, 26, 39, 23, 30, 37, 30, 20, 33, 23, 33, 31, 34, 27,  
23, 30, 32, 34, 33, 32, 30, 46, 18, 27, 24, 30, 28, 22, 25, 24, 38, 32, 27, 32, 29, 32, 25,  
26, 21, 27, 22, 42, 28, 30, 21, 45, 34, 43, 33, 25, 25, 34, 20, 44, 42, 28, 39, 15, 40, 37,  
26, 22, 30, 36, 30, 29, 33, 39, 30, 18, 33, 44, 27, 32, 28, 28, 29, 23, 31, 33, 25, 36, 32,  
29, 31, 26, 29, 34, 36, 27, 34, 17, 34, 39, 44, 30, 36, 29, 39, 26, 19, 36, 36, 28, 34, 19,  
22, 29, 22, 26, 37, 36, 29, 33, 35, 29, 28, 24, 43, 31, 25, 30, 29, 19, 20, 35, 33, 32, 32,  
26, 26, 35, 24, 33, 26, 31, 43, 27, 33, 28, 31, 32, 38, 36, 23, 35, 22, 36, 32, 42, 40, 11,  
27, 29, 34, 30, 30, 32, 31, 31, 37, 26, 22, 34, 35, 36, 35, 26, 39, 32, 29, 30, 34, 32, 31,  
27, 25, 26, 42, 39, 32, 46, 38, 28, 37, 27, 29, 24, 20, 29, 32, 37, 39, 41, 34, 23, 20, 26,  
35, 21, 23, 44, 26, 28, 40, 34, 36, 28, 35, 26, 48, 32, 28, 28, 35, 24, 40, 25, 28, 43, 20,  
39, 24, 30, 19, 26, 17, 26, 15, 26, 24, 34, 26, 35, 47, 32, 46, 48, 33, 30, 43, 21, 38, 41,  
39, 34, 41, 30, 33, 23, 24, 34, 30, 29, 32, 31, 28, 27, 28, 42, 35, 30, 31, 44, 26, 36, 40,  
40, 30, 31, 32, 32, 17, 45, 31, 29, 44, 27, 27, 22, 47, 35, 30, 31, 30, 24, 14, 21, 28, 26,  
25, 21, 19, 33, 33, 40, 22, 39, 32, 37, 20, 35, 26, 29, 34, 23, 10, 20, 31, 29, 30, 25, 35,  
33, 35, 22, 20, 33, 40, 31, 28, 31, 40, 27, 24, 35, 16, 14, 21, 23, 35, 32, 32, 31, 41, 29,  
33, 27, 39, 30, 40, 21, 31, 17, 37, 19, 31, 39, 28, 37, 29, 25, 39, 40, 24, 15, 26, 38, 35,  
24, 39, 21, 39, 27, 19, 25, 17, 38, 34, 36, 28, 21, 31, 35, 38, 36, 36, 23, 34, 28, 38, 31,  
25, 30, 21, 31, 36, 32, 28, 23, 32, 44, 33, 43, 25, 29, 40, 14, 30, 28, 30, 30, 31, 26, 15,  
37, 27, 33, 42, 36, 28, 33, 37, 29, 36, 47, 26, 23, 36, 27, 26, 42, 27, 32, 30, 43, 32, 37,  
23, 29, 30, 23, 30, 13, 26, 22, 33, 23, 27, 32, 33, 38, 41, 30, 30, 20, 22, 28, 26, 32, 24,  
23, 28, 33, 27, 37, 36, 15, 24, 30, 40, 39, 28, 22, 22, 23, 22, 39, 40, 29, 23, 38, 26, 41,  
38, 28, 24, 25, 35, 19, 23, 35, 37, 34, 34, 34, 27, 36, 30, 33, 26, 36, 26, 30, 29, 23, 27,  
43, 40, 38, 31, 35, 35, 38, 42, 42, 21, 34, 26, 45, 35, 26, 24, 29, 31, 28, 26, 30, 37, 34,  
33, 24, 26, 31, 28, 33, 29, 33, 32, 22, 38, 22, 28, 39, 20, 30, 25, 24, 30, 24, 27, 23, 29,  
36, 20, 33, 36, 26, 20, 41, 35, 21, 33, 25, 27, 21, 26, 20, 24, 35, 36, 17, 34, 17, 33, 31,

16, 15, 24, 41, 29, 18, 31, 30, 38, 28, 25, 42, 22, 30, 40, 22, 26, 28, 19, 42, 25, 36, 21, 31, 32, 32, 34, 26, 21, 31, 30, 22, 28, 25, 29, 32, 34, 41, 34, 31, 23, 24, 17, 35, 30, 39, 27, 16, 31, 31, 28, 35, 32, 39, 29, 29, 34, 23, 30, 33, 33, 23, 27, 19, 40, 24, 31, 34, 28, 22, 28, 31, 40, 22, 36, 21, 33, 33, 26, 31, 34, 25, 33, 30, 32, 40, 24, 30, 27, 36, 27, 26, 32, 28, 39, 25, 25, 34, 34, 29, 25, 24, 30, 31, 32, 34, 23, 33, 26, 32, 18, 25, 18, 37, 37, 31, 31, 30, 27, 36, 18, 38, 22, 34, 17, 31, 26, 39, 32, 28, 29, 22, 35, 32, 21, 32, 24, 33, 22, 34, 30, 20, 26, 32, 27, 6, 29, 15, 36, 30, 26, 28, 38, 33, 26, 26, 24, 43, 40, 29, 36, 27, 31, 25, 29, 33, 18, 47, 30, 24, 27, 33, 27, 29, 26, 27, 26, 33, 26, 29, 25, 40, 31, 35, 16, 33, 23, 21, 39, 25, 41, 24, 22, 13, 34, 30, 22, 31, 35, 25, 35, 41, 50, 37, 35, 22, 15, 30, 34, 37, 45, 35, 23, 27, 26, 29, 27, 34, 38, 27, 34, 24, 34, 31, 32, 27, 23, 42, 24, 30, 32, 43, 17, 30, 31, 30, 32, 9, 28, 34, 38, 32, 28, 26, 36, 27, 25, 24, 23, 34, 28, 32, 38, 35, 29, 26, 22, 26, 36, 38, 30, 30, 40, 18, 39, 22, 30, 19, 30, 26, 27, 31, 48, 33, 27, 31, 30, 30, 31, 35, 17, 37, 29, 28, 39, 32, 34, 33, 38, 24, 24, 23, 15, 26, 31, 35, 29, 25, 23, 28, 24, 16, 23, 19, 42, 34, 13, 29, 29, 33, 15, 37, 37, 38, 19, 41, 35, 34, 33, 15, 31, 27, 34, 38, 30, 30, 42, 31, 28, 32, 23, 23, 29, 41, 36, 52, 36, 40, 31, 36, 21, 28, 26, 26, 30, 22, 41, 26, 31, 36, 24, 29, 25, 32, 30, 37, 23, 34, 31, 24, 31, 28, 23, 34, 25, 30, 31, 25, 30, 36, 34, 30, 21, 35, 30]

## Решение

data\_1 = данныеизусловия

data\_2 = данныеизусловия

data\_3 = данныеизусловия

D1 = 0.0033 #данныеиззадачи 8

D2 = 0.0016 #данныеиззадачи 8

D3 = 0.0016 #данныеиззадачи 8

#находимсреднююпосещаемостьпорайонам

A1 = sum(data\_1)/len(data\_1)

A2 = sum(data\_2)/len(data\_2)

A3 = sum(data\_3)/len(data\_3)

#находимрейтинг

R1 = A1/D1

R2 = A2/D2

R3 = A3/D3

#Есливывестиих, томожемувидетьследующиерезультаты:

# 18075.252525252527, 27833.333333333332, 37355.20833333333

#Изнихвидно, чтонаибольшийрейтингимееттретийрайон.

Вусловиисказано, чтопервойцифройидетномеррайона.  
онполучилсяравен 3. Ивторойцифройрейтингсотброшеннойчастьюпослеточки  
= 37355. Значитответ 3,37355

Ответ: 3,37355

### Задача 10. Рестораниликафебыстрогопитания?

#### Условие.

Мыопределилирайон,  
вкоторомвыгоднееоткрыватьновуюточкуобщепита, теперьнужнорешить,  
какоготипабудетпредприятие: кафебыстрогопитания, рестораниликафе.

Вамдано 2 списка:

- check — показательсреднегочеказаказапокаждомузаведению.
- clients — количествопосетителейзачаспокаждомузаведению.

Определите,

какраспределяютсядолиразличныхтиповпредприятийобщественногопитания,  
существующихнатерритории, если принято считать, что:

- Кафе быстрого питания, когда:  $check \leq 250, clients \geq 20$
- Кафе:  $check > 250$  and  $check < 1000, clients \geq 10$
- Ресторан:  $check \geq 1000, clients \geq 3$

Найдите по каждой категории (bistro — Кафе быстрого питания, cafe —  
Кафе, restauraunt — Ресторан):

- количество заведений;
- среднее значение по среднему чеку (среднее значение показателя средний чек по категории);

- среднее число посетителей.

В ответе через запятую укажите среднее значение средних чеков и среднюю посещаемость для категории с наименьшей конкуренцией (меньшим количеством заведений) на исследуемой территории.

Примечание. У значений отбросьте дробную часть (например: 836,6983 получится 836).

Пример ответа: 7665,75, (где 7665 — среднее значение средних чеков, а 75 — средняя посещаемость)

### Данные.

check = [266, 427, 213, 282, 9920, 276, 240, 447, 234, 862, 684, 257, 485, 244, 263, 283, 255, 177, 238, 888, 178, 3509, 180, 409, 285, 152, 157, 197, 215, 232, 665, 154, 163, 226, 150, 276, 200, 764, 235, 5448, 179, 172, 295, 279, 195, 276, 293, 545, 230, 6444, 4444, 279, 236, 903, 6838, 226, 961, 399, 863, 419, 5063, 290, 152, 389, 997, 616, 959, 277, 926, 687, 172, 239, 494, 4660, 3343, 203, 169, 852, 290, 981, 190, 158, 170, 303, 795, 159, 819, 262, 260, 222, 219, 592, 210, 199, 223, 158, 652, 3407, 295, 258, 209, 193, 459, 157, 231, 241, 289, 181, 196, 866, 230, 368, 587, 788, 916, 1846, 2941, 351, 829, 6216, 767, 255, 685, 220, 248, 776, 521, 277, 244, 257, 282, 1935, 6662, 230, 285, 8613, 229, 208, 152, 384, 567, 6761, 285, 287, 873, 519, 166, 1370, 234, 752, 2272, 8482, 184, 3752, 262, 235, 152, 915, 182, 754, 224, 384, 876, 164, 212, 175, 769, 663, 891, 340, 1807, 285, 239, 625, 739, 494, 261, 726, 193, 803, 284, 535, 731, 990, 7076, 487, 257, 4353, 923, 378, 292, 993, 891, 5939, 214, 218, 340, 183, 325, 838, 178, 382, 285, 952, 265, 826, 526, 282, 172, 167, 151, 864, 215, 279, 999, 173, 197, 238, 203, 229, 195, 537, 180, 272, 640, 974, 464, 851, 272, 756, 595, 240, 266, 263, 168, 171, 249, 152, 864, 625, 919, 205, 707, 180, 894, 999, 2231, 780, 230, 729, 296, 692, 238, 177, 968, 183, 265, 897, 441, 160, 296, 7460, 582, 931, 216, 192, 2231, 359, 9161, 150, 212, 549, 416, 169, 187, 1858, 7999, 272, 257, 202, 158, 238, 243, 685, 4710, 271, 599, 886, 419, 790, 208, 275, 231, 427, 5137, 165, 8849, 8435, 203, 266, 430, 217, 293, 188, 172, 216, 763, 9922, 9209, 263, 297, 226, 268, 268, 286, 1170, 10057, 289, 227, 234, 174, 154, 160, 226, 463, 809, 530, 298, 242, 698, 540, 414, 10029, 9406, 259, 174, 223, 364, 3749, 519, 505, 7518, 281, 772, 399, 774, 159, 194, 184, 279, 253, 1680, 7249, 551, 8087, 278, 373, 694, 289, 5027, 8635, 301, 267, 358, 298, 281, 262, 172, 407, 265, 290, 693, 781, 528, 594, 680, 233,

943, 169, 607, 419, 428, 166, 160, 7018, 283, 164, 415, 726, 8835, 987, 176, 546, 441, 214, 750, 209, 236, 532, 736]

```
clients = [17, 18, 22, 12, 3, 14, 22, 10, 21, 13, 17, 17, 18, 27, 16, 16, 11, 23, 25, 16,
22, 4, 24, 13, 11, 20, 22, 20, 27, 23, 18, 20, 25, 21, 26, 14, 21, 12, 22, 5, 23, 22, 18,
16, 24, 18, 10, 19, 21, 4, 5, 18, 20, 17, 5, 27, 14, 17, 19, 16, 5, 10, 23, 19, 10, 17, 17,
17, 15, 18, 22, 24, 18, 5, 4, 22, 21, 18, 14, 17, 22, 21, 25, 10, 15, 22, 15, 17, 16, 21,
23, 19, 20, 27, 21, 21, 18, 4, 11, 18, 22, 24, 15, 27, 22, 21, 15, 27, 24, 11, 26, 11, 18,
16, 10, 3, 3, 13, 15, 3, 13, 16, 17, 27, 20, 18, 10, 14, 24, 16, 15, 3, 3, 21, 14, 4, 23, 24,
27, 11, 14, 5, 10, 12, 17, 19, 21, 4, 22, 17, 4, 3, 24, 5, 13, 22, 22, 12, 20, 15, 27, 15,
14, 26, 22, 23, 13, 18, 13, 16, 4, 19, 22, 12, 18, 13, 17, 17, 23, 14, 10, 19, 10, 16, 4,
10, 17, 3, 17, 14, 16, 11, 10, 3, 26, 24, 19, 27, 19, 18, 22, 18, 10, 19, 17, 14, 19, 16,
25, 26, 25, 19, 21, 10, 12, 25, 25, 24, 20, 24, 25, 11, 22, 14, 14, 19, 19, 19, 17, 17, 10,
20, 12, 15, 26, 22, 21, 23, 17, 19, 13, 24, 18, 25, 15, 13, 5, 17, 26, 16, 17, 14, 22, 27,
16, 27, 15, 12, 13, 27, 19, 4, 17, 10, 27, 25, 4, 14, 3, 22, 22, 12, 11, 21, 23, 5, 5, 12,
16, 25, 22, 21, 21, 14, 5, 17, 11, 16, 17, 17, 22, 17, 20, 12, 5, 24, 3, 4, 22, 17, 19, 20,
17, 20, 22, 20, 18, 5, 4, 18, 17, 23, 17, 10, 17, 3, 4, 11, 23, 20, 27, 24, 20, 26, 14, 10,
17, 11, 27, 14, 19, 16, 3, 4, 16, 21, 24, 16, 3, 13, 19, 3, 12, 17, 11, 12, 25, 25, 20, 14,
17, 3, 5, 14, 4, 14, 19, 11, 17, 3, 3, 18, 17, 18, 13, 17, 18, 24, 12, 12, 17, 12, 19, 17,
12, 15, 21, 17, 24, 14, 15, 19, 23, 22, 3, 17, 24, 17, 15, 4, 13, 21, 14, 16, 22, 18, 22,
20, 16, 17]
```

## Решение

check = данныеизусловия

clients = данные из условия

#переменные для подсчета по всем районам

#type\_1 - количество, check1 - сумма чеков, clients1 - сумма клиентов

type\_1 = 0

check1 = 0

clients1 = 0

type\_2 = 0

check2 = 0

clients2 = 0

type\_3 = 0

check3 = 0

clients3 = 0

```
for i in range(len(check)):
```

```
if check[i]<=250 and clients[i]>=20:
    type_1+=1
    check1+=check[i]
    clients1+=clients[i]
elif check[i]>250 and check[i]<1000 and clients[i]>=10:
    type_2+=1
    check2+=check[i]
    clients2+=clients[i]
elif check[i]>=1000 and clients[i]>=3:
    type_3+=1
    check3+=check[i]
    clients3+=clients[i]
```

**#выводим общее количество заведений, средний чек и среднее число клиентов по каждой категории**

```
print(type_1,check1/type_1,clients1/type_1)
print(type_2,check2/type_2,clients2/type_2)
print(type_3,check3/type_3,clients3/type_3)
```

Результат, следующий:

```
135 197.8074074074074 23.081481481481482
215 530.6 15.130232558139534
50 5775.26 3.9
```

Наименьшее количество заведений имеет третья категория “ресторан”. Отбрасываем у значений дробную часть и получаем ответ 5775,3(где 5775— среднее значение средних чеков, а 3— средняя посещаемость)

**Ответ:5775.3**

## Задачи командного этапа

### Легенда.

В компании AltFuture стартует новый проект! Создан отдел неЛинейного счастья. Отдел ведет работу по исследованию условий создания комфортной городской среды обитания. Градоначальники, заботящиеся о достойном жизнеобеспечении своих граждан, стали обращаться в новый отдел за советом и анализом ситуации.

Однажды в отдел неЛинейного счастья пришел мэр города Энска. Современный город — это экосистема. Мэр и администрация Энска стараются создавать в нем наиболее благоприятные условия для жизни. Junior-стажерам необходимо провести исследование, результатом которого будет изучение настроения жителей различных кварталов города и рекомендации по проектированию или *реинжинирингу*<sup>1</sup> (реконструкции) объектов городской инфраструктуры.

### Вариативность.

Энск несуществующая территория: карта города, данные о погодных условиях, природе и рельефе, объектах жизнеобеспечения, жителях, отзывы жителей разных районов, сайт города Энск – являются *синтетическими*<sup>2</sup> (искусственными, сгенерированными на компьютере). Для каждого финала генерируются новая матрица данных.

### Генеральная задача командного этапа.

Познакомиться с информацией, размещенной на сайте города Энск. Прочитать вводную статью мэра, опираясь на прочитанный текст выделить

---

<sup>1</sup>*Реинжиниринг* - фундаментальное переосмысление и радикальное перепроектирование для достижения максимального эффекта производственно-хозяйственной и финансово-экономической деятельности.

<sup>2</sup>*Синтетические (данные, объекты)* – это искусственные данные, а точнее «любые данные, применимые к данной ситуации, которые не получены прямым измерением».



проблемы города, как среды. Выбрать район для исследования. Используя сайт, собрать данные (*парсинг*, *скрапинг*<sup>3</sup> данных) об объектах городской инфраструктуры выбранного для исследования квартала. Провести анализ собранных данных. Выяснить, настроение жителей города: что делает счастливыми горожан, какие проблемы городской среды обитания их волнуют и вызывают недовольство. Построить графики, иллюстрирующие результаты анализа данных. Разработать рекомендации по устранению проблем в том или ином районе. Предложенные изменения должны укладываться в определенную сумму условных денежных единиц, выделяемых мэрией Энска, а некоторые решения предполагать вариативность в зависимости от разных параметров. Оформить отчет об исследовании в виде интерактивной публикации – вычислительного эссе или представить презентацию с обзором выявленных проблем и методами их решения

Кроме данных о городе Энск, размещенных на веб-ресурсе, команде Junior-стажеров предоставляются *датасеты*<sup>4</sup> о различных городах нашей страны, используя которые, можно отыскать всех двойников города Энск по различным параметрам: погодные условия, рельеф, интересы и занятия жителей, плотность населения, любимые места отдыха горожан, трудовая деятельность и т. п. В каждом из этих городов двойников есть ключ к решению наиболее острых проблем города.

### *Задача 1. Получение данных в Google Collab.*

Основная цель первой задачи – знакомство участников со способами работы с инструментами GoogleCollab.

#### **Условие задачи.**

---

<sup>3</sup>*Веб-скрапинг (парсинг) данных – это извлечение, сбор любой открытой информации с веб-сайта, позволяет быстро получить и сохранить структурированные данные веб-ресурса. Веб-скрапинг может быть сделан вручную пользователем компьютера, однако термин обычно относится к автоматизированным процессам, реализованным с помощью кода, который выполняет запросы на целевой сайт.*

<sup>4</sup>*Датасеты - структурированные выборки данных по различным тематикам*

Необходимо скопировать интерактивную тетрадку шаблон Google Collab к себе на диск по инструкции из предоставленного файла. Инструкция для задачи представлена в приложении 1.

### *Задача 2. Анализ статьи мэра*

Эта задача также относится к вводным и меняется в зависимости от команды и этапа финала. Каждая команда получает ссылку на комплекс входных данных, оформленный в виде страницы некоторого сайта. Пример оформления входных данных можно увидеть, если перейти по ссылке ниже.

#### **Пример входных данных.**

<https://yupest.github.io/nti/site/final1/district2.html>

#### **Описание задачи.**

Создайте копию на своем диске предоставленной интерактивной тетради с переменными в GoogleCollab, для итогового эссе, правильно настройте ее для командной работы.

Проанализируйте обращение мэра на сайте Дистрикта, выявите по данным мест в Дистрикте (ниже статьи мэра) какая именно проблема существует в районе из описанных в выступлении мэра.

Сформулируйте несколько предположений(гипотез), с чем связана данная проблема в Дистрикте, на основе информации, предоставленной на сайте данных.

Запишите свои предположения в рабочую тетрадь.

В ответе к данному шагу укажите номер Дистрикта, данные которого представлены на сайте.

Примечание: номер можно определить по выявленной проблеме района.

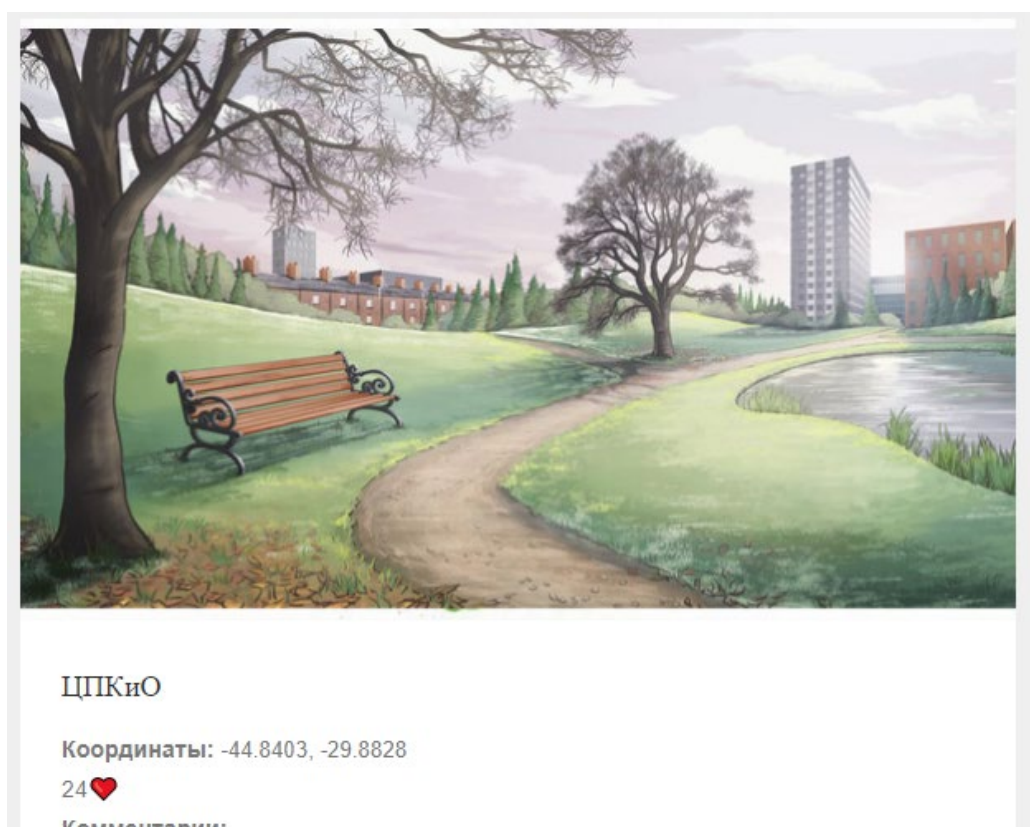
#### **Решение**

Проанализировав статью мэра и данные с сайта, можно определить номер дистрикта.

Про первый Дистрикт мэр говорит: “В одном из районов (District 1) города построен современный развлекательный центр «Остров пиратов» ...” но такого места на сайте нет, соответственно это не наш дистрикт.

Про третий Дистрикт сказано: “В районе District 3, построен новый жилой квартал «ГотемСити» ...”, но он тоже не представлен на сайте, что позволяет сказать, что это тоже не анализируемый дистрикт.

Исходя из наблюдений, можно сказать, что Дистрикт 2 является районом, представленным на сайте, это подкрепляется также тем, что на сайте есть место, о котором говорит мэр в своей статье (рис. 43).



*Рис. 43. Искомый район города*

## Выводы

Прочитав обращение мэра и комментарии пользователей сайта, мы можем сформулировать главную проблему, связанную с нашим районом, которая волнует жителей на данный момент:

- что не так с ЦПКиО? Почему он не может по-прежнему собирать огромное количество людей?

На основе тех же комментариев можем сделать несколько предположений (гипотез) о причинах возникновения проблемы:

- наиболее часто встречающаяся жалоба – доступность парка;
- отсутствие развития парка, как места отдыха горожан;
- также жалуются на неухоженность данного объекта;

Важной ремаркой будет то, что проблемы, которые были перечислены выше, присутствовали ещё в 2017 году, а последний комментарий, относящийся к этим проблемам, появился в 2020 году.

При этом стоит отметить, что горожан полностью устраивает местоположение данного городского объекта: "Хочется, чтобы парк развивался, местоположение огонь!".

### **Ответ.**

Дистрикт 2

### *Задача 3. Парсинг сайта.*

Сайт, пример которого приведен во второй задаче является основой для множества последующих задач и, соответственно, используется как набор входных данных.

### **Входные данные.**

Сайт для парсинга (см. пример по ссылке из задачи 2, стр. 53).

### **Условие задачи.**

Сайты являются ценным источником информации для исследователей, потому что содержат не только данные о местах города, но и пользовательские данные.

Однако, для того чтобы получить эту информацию в удобном для анализа виде и сэкономить время сбора информации, необходимо пользоваться специализированными инструментами.

Вам предлагается воспользоваться инструментами парсинга (Google Colab и библиотеки `bs4` и `requests`): извлечь информацию из содержимого элемента `html` или значений атрибутов, обращаясь к ним с помощью функции `find_all()`, на вход которой прописаны теги и атрибуты (если атрибуты есть) и сделать преобразование типов в соответствии с типом информации.

При парсинге данных определите у элемента "категория места" значения элементов, в котором она расположена:

- имя тега;
- имя атрибута;
- значение атрибута;
- и последним словом одно из двух:
  - имя атрибута, если значение "категория места" находится в значении атрибута;
  - слово "text", если значение категорий находится в содержимом элементе.

В ответе укажите путь к значениям категории мест в порядке, перечисленном выше, через точку с запятой без пробелов.

Например, если бы необходимо было получить названия мест, то ответ выглядел бы так: `h2,name,locations,text`.

Примечание. Какие-то значения могут остаться незаполненными, если их нет. В ответе обязательно соблюдайте порядок значений в соответствии с

представленным форматом! Пробелов рядом с запятыми быть не должно, но они могут присутствовать между словами одного значения, если это значение имеет несколько слов, например, значение атрибута из 2 слов: `mycategory` - запишется через пробел.

### Решение

```
from bs4 import BeautifulSoup
import requests
```

```
URL = "https://yupest.github.io/nti/site/final1/district2.html"
html = requests.get(URL).text
soup = BeautifulSoup(html, 'lxml')
categories = [r for r in soup.find_all(attrs={"name": "category"})]
categories[0]

```

**Ответ:** `img,name,category,alt`

### Задача 4. Парсинг для извлечения данных

Задача является естественным продолжением предыдущей задачи.

#### Входные данные.

Сайт для парсинга из задачи 2.

#### Условие задачи

Сайты являются ценным источником информации для исследователей, потому что содержат не только данные о местах города, но и пользовательские данные.

Однако, для того чтобы получить эту информацию в удобном для анализа виде и сэкономить время сбора информации, необходимо пользоваться специализированными инструментами. Вам предлагается воспользоваться инструментами парсинга (Google Collab и библиотеки `bs4` и `requests`): извлечь информацию из содержимого элемента `html` или значений атрибутов,

обращаясь к ним с помощью функции `.find_all()`, на вход которой прописаны теги и атрибуты (если атрибуты есть) и сделать преобразование типов в соответствии с типом информации.

Информация, которую необходимо получить и ее тип:

- категория места - строковый тип данных (`str`) - как вы можете заметить, на сайте пользователям не представлена категория мест, ее можно узнать только просмотрев код страницы
- отзывы пользователей по каждому месту - тип данных строковый (`str`)
- оценка отзывов (`rating`) - целочисленный тип данных (`int`)

Для удобства поиска путей (тегов и атрибутов) к информации вы можете просмотреть код страницы по ссылке в 1 шаге или в переменной `soup`.

Тексты отзывов и их оценка вам необходима для реализации последующих этапов. Однако, является не обязательным условием. Если парсинг этой информации не получился, то в дальнейшем вы сможете вручную скопировать нужную информацию с сайта, но такой подход, возможно, увеличит время исполнения задачи.

Решение

```
locations = [r.text for r in soup.find_all('h2')]
locations = locations[2:]
reviews = [r.text for r in soup.find_all("p", {"class": "review"})]
locations_coordinates = []
lattitudes = [float(r.text) for r in soup.find_all('span', {'name': 'latitude'})]
longitudes = [float(r.text) for r in soup.find_all('span', {'name': 'longitude'})]
for i in range(len(lattitudes)):
    locations_coordinates.append([lattitudes[i], longitudes[i]])
likes = [int(r.text) for r in soup.find_all('span', {'name': 'count_likes'})]
density = [int(el.div['style'][7:][: -1]) for el in soup.find_all('div', {'class': 'pipe line'})]
```

После выполнения кода получаем массивы необходимых данных (рис. 20), а именно:

- категория места: строковый тип данных (str). Как вы можете заметить, на сайте пользователям не представлена категория мест, ее можно узнать только просмотрев код страницы;
- отзывы пользователей по каждому месту: тип данных строковый (str);
- оценка отзывов (rating): целочисленный тип данных (int)

```
[ 'Дистрикт №2', 'Обращение мэра Дистрикта №2', 'Домино', 'ЦПКиО', 'Тридцатье Королевство',
[
['Мы живем тут уже три года. Транспортная доступность здесь отличная. На 19 автобусе, который
[[-16.2991, -39.0234], [-44.8403, -29.8828], [74.4964, 6.3281], [56.17, 44.2969], [44.5905,
[87, 24, 67, 78, 60, 97, 50, 65, 45, 48, 88, 56, 93, 84, 92]
[30, 35, 30, 29, 17, 57, 57, 8, 5, 8, 6, 5, 20, 20, 36, 67, 67, 55, 55, 22, 22, 69, 57, 52, 5
```

*Рис. 44. Пример полученных данных.*

### *Задача 5. Анализ зеленых насаждений городской среды*

#### **Входные данные.**

Фотография района с квадрокоптера (первое фото в слайдере в оглавлении сайта вашего дистрикта).

#### **Условие задачи.**

На основе фотографии района с квадрокоптера определить отношение зеленых насаждений к другим объектам.

Для анализа изображения воспользуйтесь сервисом по определению цветов с картинки (например: [https://hysy.org/get\\_colors](https://hysy.org/get_colors)):

- 1) установите максимальный размер кисти;
- 2) выберите 8 цветов палитры;
- 3) скачайте с сайта своего Дистрикта фотографию с квадрокоптера и загрузите ее в сервис;
- 4) сложите процентные значения зеленых цветов, отличные от 0%, получившиеся при анализе фото сервисом;
- 5) постройте диаграмму пирог, показывающую отношение зеленых насаждений района к "не зеленым". Отобразите результат в своем эссе;



- б) на основе получившейся диаграммы сделайте вывод об уровне озеленения города;
- 7) оформите выводы в эссе, прикрепите к ним диаграмму.

**Решение.**

Фотография района Дистрикта выглядит следующим образом (рис. 45):



*Рис. 45. Фотография района*

При загрузке изображения на сервис определения цветов с нужными настройками получаем следующий результат (рис. 46):

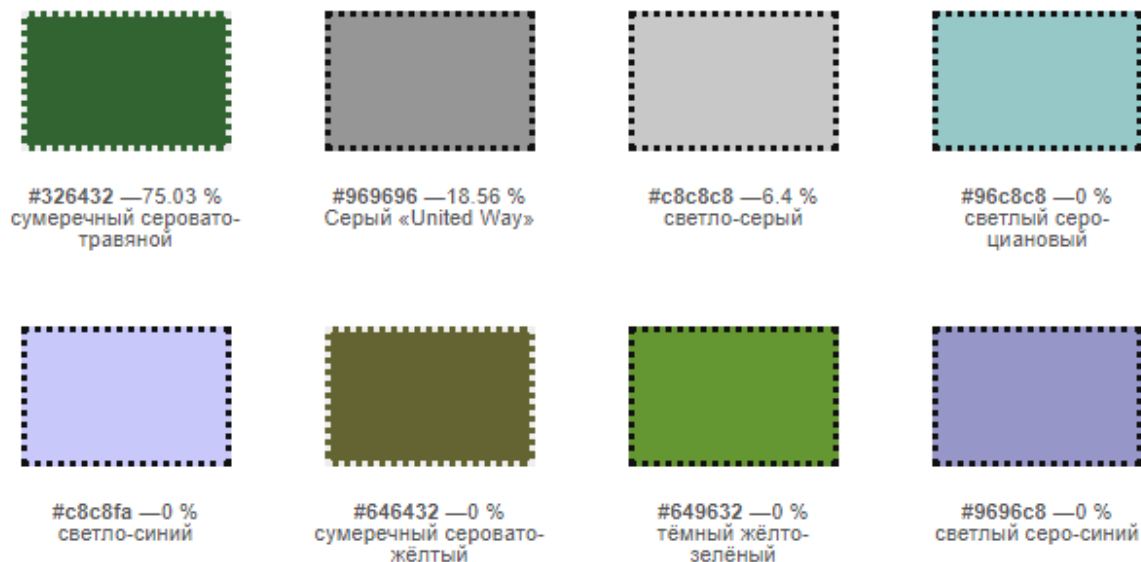


Рис. 46. Результат анализа цветов

Исходя из результата можно сделать следующий вывод: есть только один зеленый цвет, содержание которого более 0%. Это первый цвет (рис. 47).

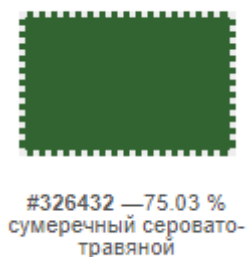


Рис. 47. Зеленый цвет, содержание которого больше 0

На результате видно, что его содержание 75%, соответственно нам нужно построить круговую диаграмму, где зеленых насаждений будет 75%, а “не зеленых” - оставшиеся 25%. Воспользуемся тетрадкой Google Collab и напишем код для ее создания.

```
import matplotlib.pyplot as plt
```

```
green = 75.03
other = 24.96
values = [green, other]
```

```
labels = ["зеленый " + str(green) + "%", "незеленый " + str(other) + "%"]
plt.pie(values, labels=labels)
plt.title("District 2")
plt.show();
```

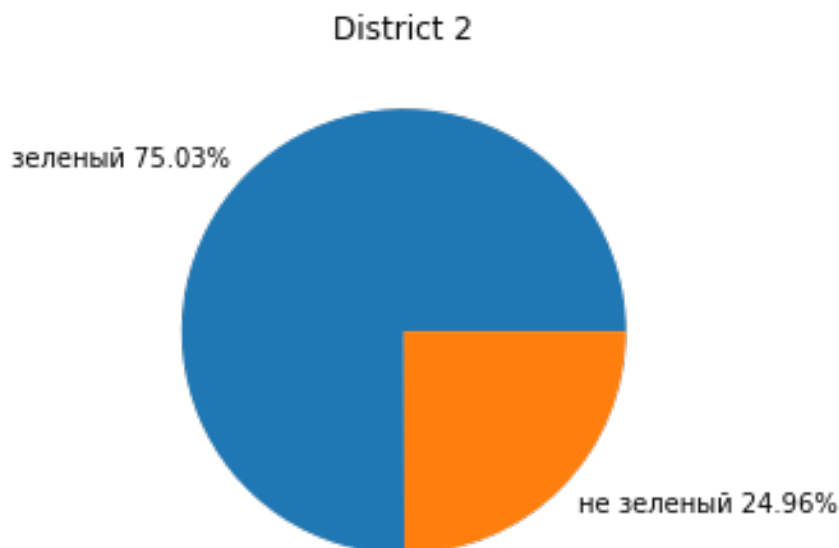


Рис. 48. Результат работы программы

### Вывод

Из представленной в слайдере сайта картинке мы получаем соотношение зеленых насаждений к “не зеленым”. На основании данного соотношения можно сделать вывод, что уровень озеленения в Дистрикте довольно хороший (3/4 от общего), этот показатель гораздо лучше показателя в 2019 году, что несомненно говорит о том факте, что власти города прислушиваются к пожеланиям горожан.

### Задача 6. Анализ зеленых насаждений (нахождение аналога)

Задача не может быть решена без результатов предыдущей задачи. Такой подход к построению серии последовательных задач позволяет косвенно контролировать выбор командами решаемых задач, способствует концентрации внимания.

### Входные данные.

Диаграмма-пирог с результатом анализа насаждений в Дистрикте (задача 5, рис. 48)

**Условие задачи.**

Найдите город-аналог (рис. 49) со схожим процентным содержанием зеленых насаждений по сравнению с вашим районом. В ответе укажите название найденного города.

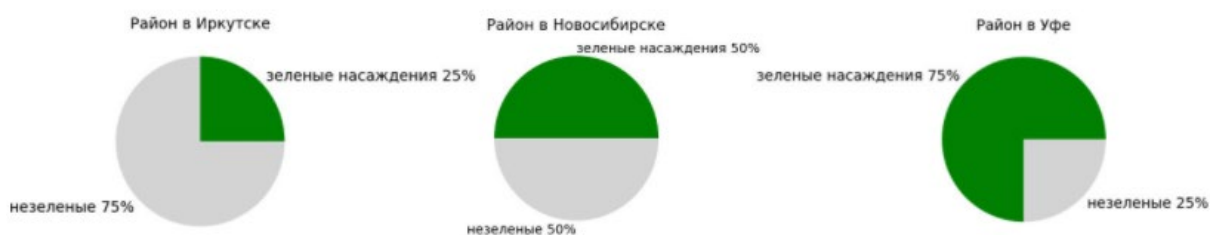


Рис. 49. Города-аналоги предложенного района.

**Решение.**

Для решения этой задачи возьмем диаграмму, получившуюся в задаче 5 (рис. 24)

На основе диаграммы и сравнения её с образцами из условия, можно сделать вывод, что наиболее схожее процентное содержание зеленых насаждений в городе Уфа. Значения почти равны и отличаются лишь на сотые доли.

**Ответ.**

Уфа

*Задача 7. Интересы Дистрикта*

**Входные данные.**

Набор исходных данных представляет собой заранее подготовленный dataset в отдельном файле, который участники могут посмотреть и скачать через электронную среду. Формат файла: \*.csv, что позволяет разработать не только программное решение задачи, но и найти решение с помощью MicrosoftExcel.

### Условие задачи.

Проанализируйте интересы исследуемого района города N на основе данных, представленных в файле интересов Дистрикта.

Найдите 5 типов интересов, которые чаще всего встречаются в Дистрикте. Постройте столбчатый график интересов.

### Решение

```
import pandas as pd
import collections
import matplotlib.pyplot as plt

def get_top(filename, most_common = 5):
    df = pd.read_csv(filename)
    column = df.columns[0]
    c = collections.Counter()
    for word in df[column]:
        c[word] += 1
    c[column] += 1
    answer = []
    for place in c.most_common(most_common):
        answer.append(place[0])
    return answer
origin = get_top('interest_nsk.csv')
print(';'.join(origin))
leisure = []
leisure_weight = []
for place in c.most_common(5):
    leisure.append(place[0])
    leisure_weight.append(place[1])
plt.figure()
plt.bar(leisure, leisure_weight)
plt.show()
```

Результат работы программы представлен на рис. 50.

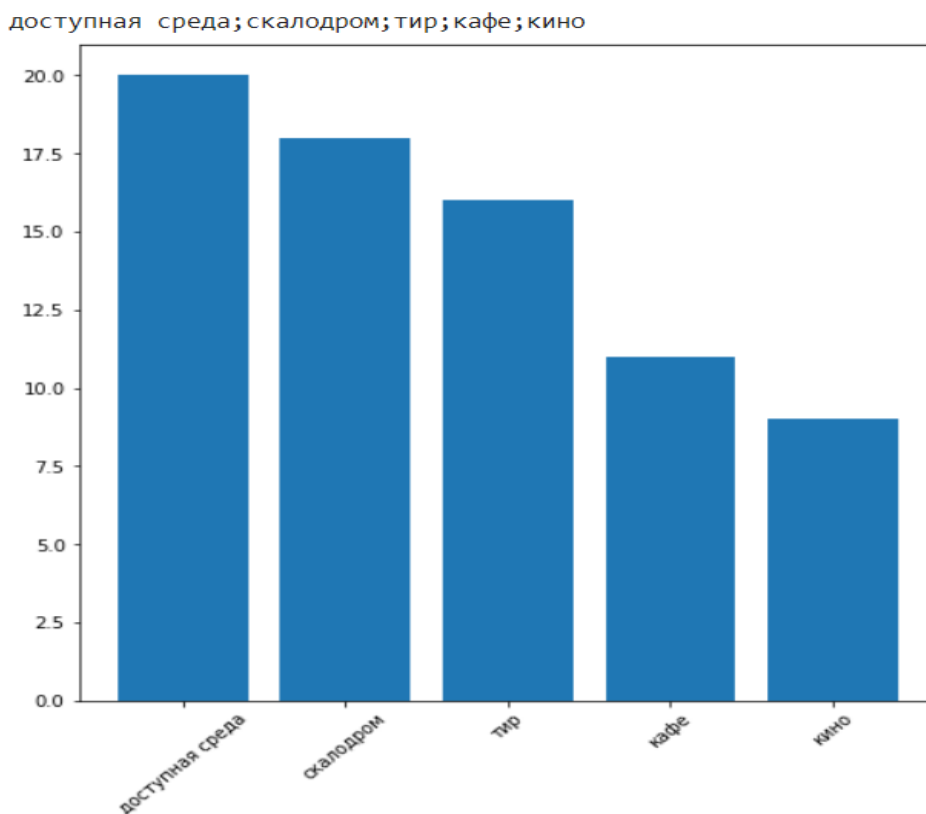


Рис. 50. Результаты анализа интересов жителей района

На графике можно увидеть, что жителям анализируемого Дистрикта наиболее интересны: доступная среда, скалодром, тир, кафе и кино

#### Задача 8. Интересы Дистрикта (поиск аналога)

##### Входные данные.

График интересов Дистрикта (задача 7, рис. 50), файл интересов Дистрикта, файлы интересов трех других городов, представленные отдельными dataset-файлами.

##### Условие задачи.

Проанализируйте данные в файлах по интересам трех других городов. Составьте рейтинг из 5 приоритетных интересов жителей.

Сравните результаты интересов вашего Дистрикта и получившиеся топы интересов трех других городов. Определите город(город-аналог), наиболее похожий по интересам на ваш Дистрикт.

В ответе к данному шагу укажите три приоритетных часто встречающихся интереса в городе-аналоге через, без пробелов. Например: санки;кофе;кактусы

### Решение задачи.

```
import pandas as pd
import collections

def get_top(filename, most_common = 5):
    df = pd.read_csv(filename)
    column = df.columns[0]
    c = collections.Counter()
    for word in df[column]:
        c[word] += 1
    c[column] += 1
    answer = []
    for place in c.most_common(most_common):
        answer.append(place[0])
    return answer
origin = get_top('interest_nsk.csv')
others = []
closest = 0
closest_index = 0
for i in range(1, 4):
    others.append(get_top('interest_{}.csv'.format(i)))
    cnt = 0
    for place in origin:
        if place in others[-1]:
            cnt += 1
    if cnt > closest:
        closest_index = i - 1
        closest = cnt
    cnt = 0
print('.'.join(others[closest_index][:3]))
```

**Ответ.**

доступная среда; тир; скейтпарк

### *Задача 9. Публикация аналитических выводов*

#### **Входные данные.**

График интересов Дистрикта, полученный при решении задачи 7.

#### **Условие задачи.**

На основе полученного аналитиком команды графика интересов вашего Дистрикта сделайте выводы по интересам жителей вашего района и опубликуйте их в тетради вместе с графиком.

#### **Решение.**

На графике интересов Дистрикта (Рис. 50) отображены 5 наиболее популярных мест среди жителей района.

#### **Вывод.**

Проанализировав график интересов жителей Дистрикта, мы можем заметить следующее.

Самым популярным интересом является “доступная среда” — это может свидетельствовать о вовлеченности граждан в дела города и неравнодушие к людям с ограниченными возможностями, а также о том, что в городе доступная среда не сформирована.

Интересы “скалодром”, “тир” и “кино” относятся к развлекательным, соответственно жителям не хватает именно таких мест, это следует учесть при реконструкции города и запланировать подобные элементы инфраструктуры.

Интерес к “кафе” говорит о том, что в городе мало мест для питания, это тоже стоит учитывать при формировании новой инфраструктуры Дистрикта.

### *Задача 10. Визуализация мест*



### **Входные данные.**

Карта дистрикта, координаты мест, названия точек, категория точек, число лайков точки, группированные в dataset-файлы, интерактивная тетрадь с переменными.

### **Условие задачи.**

Создайте визуализацию всех точек района по их координатам на карте, такие точки называют геометками. При визуализации геометок отобразите популярность и тип каждой метки.

Для реализации в Google Collab: чтобы узнать, в каких переменных содержатся нужные данные, не забывайте пользоваться функцией `find_vars(17)` в вашей интерактивной тетрадке. Создайте визуализацию геоточек (по координатам: широта и долгота) на карте с ранжированием по популярности и типу мест. Для этого используйте подложку - изображение с сайта вашего Дистрикта (третье фото в слайдере в оглавлении страницы). Тип места — это категория (вы можете использовать переменную результата парсинга или взять значение из готовых данных). Популярность места определяется числом лайков. Узнайте уникальное число категорий, чтобы решить задачу классификации категорий на карте.

Дизайнер создает иконки для маркеров по числу уникальных категорий, различающихся по цвету, форме или цвету и форме одновременно. Для этого загрузите иконки на хостинг картинок. Скопируйте ссылки на иконки и присвойте их в соответствии с нужными категориями мест. Размер иконки должен показывать популярность места. А значит, нужно задать значение лайка - размеру маркера. Отобразите карту в своем ноутбуке Google Collab.

Альтернативное решение без программирования: Вам предложено изображение карты с красными маркерами - местами (шаблон карты меток).

Чтобы узнать какая точка к какому месту относится - рассмотрите интерактивную карту (последний слайд) в слайдере. При клике на точки отображается названия мест. Изучите таблицу мест (файл с 3 столбцами: название мест, их категория и количество лайков) и определите количество уникальных категорий. В графическом редакторе нанесите поверх этого изображения ваши иконки, отличающиеся для каждой категории по форме, цвету или форме и цвету одновременно. Размером иконки покажите популярность места. Готовую карту поместите в Google Collab.

В ходе этого задания определите категорию и название самого популярного места.

Формат ответа: категория-название. Через тире, без пробелов, при этом, если одна из частей ответа состоит из 2х и более слов пробел между ними сохраняется. Пробел при этом должен быть один!

Например:кафе-улыбка радости парк-динамо развлекательный центр-мир детства кафе-кафе радость

### **Решение.**

Используя все входные данные о Дистрикте и предварительно отрисовав иконки для каждой категории места получаем следующую карту (рис. 51):



Рис. 51. Карта района с метками

Из нее явно сказать, какая из точек наиболее популярно нельзя. Поэтому мы заново просматриваем количество «лайков» у каждого места. В жилом комплексе «Ласточка» 97 «лайков» что является наибольшим значением среди всех выделенных жителями мест.

**Ответ:** жилой комплекс-ласточка.

*Задача 11. Визуализация мест.*

**Входные данные.**

Карта Дистрикта с геометками (задача 10, рис. 51).

**Условие задачи.**

Разместите получившуюся при решении задачи 10 карту Дистрикта в эссе (тетрадь Google Collab), сделайте выводы о корректности расстановки геометок.

### **Ответ.**

Исходя из этой карты, мы видим, что ЦПКиО имеет самую маленькую геометку, хотя его местоположение достаточно хорошее.

### *Задача 12 Частотный анализ отзывов.*

#### **Входные данные.**

Данные по отзывам мест (входной dataset задачи 4).

#### **Условие задачи.**

Альтернативный способ получения данных по отзывам мест: если вы пропустили задачу 4 или не смогли её решить, можно собрать все отзывы с сайта (см. задачу 2) вручную, т. е. скопировать тексты отзывов.

Пользователи на сайте делятся своим мнением в виде отзывов о различных местах Дистрикта.

Сгенерируйте облако слов по всем отзывам, чтобы распределить слова отзывов по частоте их упоминания. Исключите такие слова как: место, здесь, очень. Возьмите для анализа отзывов только слова длиной от 4 букв и больше.

Определите ключевое существительное, которое пользователи используют чаще всего. Если самое упоминаемое слово не является существительным, исключите его из выборки и сгенерируйте облако слов еще раз. Укажите существительное в ответе, сохраняя форму слова.

#### **Решение.**

```
from bs4 import BeautifulSoup
import requests
```

```
def isNum(string):
    digits = [i for i in range(10)]
```



этого мы можем заметить, что наиболее большим словом из оставшихся является: площадка

**Ответ:** площадка

### *Задача 13. Семантический анализ отзывов*

#### **Входные данные.**

Данные по отзывам мест (задача 4, или скопированные с сайта вручную в один файл), данные по оценке отзывов (задача 4)

#### **Условие задачи.**

Альтернативный способ получения данных по отзывам мест: если вы пропустили задачу 4 или не смогли её решить, можно собрать все отзывы с сайта вручную, т. е. скопировать тексты отзывов. Затем добавить во втором столбце оценку по каждому отзыву, если у вас не вышло извлечь эти данные парсингом сайта.

Семантический анализ подразумевает распределение отзывов пользователей по настроению, которое бывает негативным и позитивным.

Отзывы имеют оценку пользователей. Считается, что рейтинг от 3 и ниже определяет отзыв как негативный, а позитивные имеют оценку 4 или 5.

Распределите отзывы пользователей с сайта дистрикта города N по настроению.

Постройте облака слов к каждому из настроений, чтобы найти главную позитивную сущность (существительное слово) и негативную. Исключите из выборки слово "место, очень, здесь". Возьмите слова от 4 букв.

В ответе напишите эти слова в виде: негативное слово;позитивное слово.

Слова указываются через «;» и соблюдается их порядок (сначала негативное, затем позитивное). Ответ вводится без пробелов.Например: мусор;скамейка

**Решение.**

```
positive = []
negative = []
locations_blocks = soup.find_all("div", {"class": "col-md-4 project-item mix"})
user_ratings = [r.text for r in soup.find_all("span", {"class": "rating"})]
for i in range(len(user_ratings)):
    if int(user_ratings[i]) > 3:
        positive.append(reviews[i])
    else:
        negative.append(reviews[i])
positive_frequency_list = {}
negative_frequency_list = {}
symbols_to_be_removed = [',', '!', ')', ':', ';', '?', '-', '<<', '>>', '\r\n', '"]
for j in positive:
    review = j
    for i in symbols_to_be_removed:
        review = review.replace(i, "")
    for string in review.split(" "):
        string = string.lower()
        if len(string) >= 4 and string.lower() != "очень" and string.lower() !=
"место" and string.lower() != "здесь" and not isNum(string):
            if string in positive_frequency_list.keys():
                positive_frequency_list.update({string:
int(positive_frequency_list.get(string)) + 1})
            else:
                positive_frequency_list.update({string: "1"})
for j in negative:
    review = j
    for i in symbols_to_be_removed:
        review = review.replace(i, "")
    for string in review.split(" "):
        string = string.lower()
        if len(string) >= 4 and string.lower() != "очень" and string.lower() !=
"место" and string.lower() != "здесь" and not isNum(string):
            if string in negative_frequency_list.keys():
                negative_frequency_list.update({string:
int(negative_frequency_list.get(string)) + 1})
            else:
                negative_frequency_list.update({string: "1"})
positive_words = ""
```





*Задача 14. Анализ отзывов (завершение)*

**Входные данные.**

Облако часто упоминаемых слов (задача 12), Облако слов позитивного настроения (задача 13), Облако слов негативного настроения (задача 13)

**Условия задачи.**

По результатам анализа отзывов добавьте в эссе получившиеся облака слов и выводы, которые по итогу решения данных задач можно сформулировать

**Решение.**

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt

wordcloud = WordCloud(background_color='white').generate(frequency_words)
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
wordcloud = WordCloud(background_color='white').generate(positive_words)
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
wordcloud = WordCloud(background_color='white').generate(negative_words)
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```



Рис. 54. Результат работы программы, составляющую комплекс облаков слов.

## Выводы.

Не беря в расчет наречия и рассматривая только существительные, можно четко проследить корреляцию наиболее встречающегося негативного слова "парк" с отзывами пользователей об этом месте. Наиболее встречающееся позитивное слово "площадка" в совокупности с прилагательным "детская" наталкивает на мысль, что у многих, кто оставлял позитивные отзывы, есть дети, что позволяет уже дальше делать выводы по поводу среднего возраста, проживающего в городе населения, а также оценить потребности именно для этих возрастных категорий граждан.

### Задача 15. Посещаемость мест

#### Входные данные.

Посещаемость мест по дням недели. Задача для аналитика команды.

#### Условие задачи.

На основе данных посещаемости мест с сайта необходимо найти самое посещаемое место в вашем Дистрикте.

Данные представляют собой процент посещаемости каждого места в вашем районе по дням недели.

Постройте столбчатую диаграмму по среднему значению посещаемости мест в Дистрикте. Найдите самое посещаемое место в вашем Дистрикте города N.

В ответе запишите название найденного места.

#### Решение

```
density_by_object = []
for i in range(0, len(locations) * 7, 7):
    density_by_object.append(density[i:i+7])
average_density = [round(sum(i) / 7, 2) for i in density_by_object]
plt.bar(locations, average_density, width=.5)
plt.gca().set_xticklabels(locations, rotation=30, horizontalalignment='right')
plt.title("Средняя посещаемость объектов", fontsize=22)
plt.ylabel("чел/день")
plt.ylim(0, max(average_density))
plt.show()
print("Самое посещаемое место",
      locations[average_density.index(max(average_density))])
print(max(average_density))
```



Рис. 55. Диаграмма средней посещаемости объектов

На графике видно, что самым посещаемым местом является - Галерея Детроит.

**Ответ:** Галерея Детроит

*Задание 16. Посещаемость мест*

**Входные данные.**

Столбчатая диаграмма по среднему значению посещаемости мест в Дистрикте (задача 15, рис. 55)

**Условия задачи.**

Оформите результаты анализа посещаемости мест Дистрикта в эссе, добавьте в него получившуюся диаграмму и сделайте выводы.

**Решение.** Разместить в тетради диаграмму (рис. 55).

## Выводы

Самым непопулярным местом является парк культуры и отдыха, о котором и говорит мэр. Популярными местами являются: галерея Детроит, изумрудный город, Тридевятое королевство.

Если посмотреть на отзывы этих мест, то видна взаимосвязь между отзывами жителей Дистрикта и средней частотой посещения того или иного места, что говорит о полезности отзывов для сбора причин популярности и антипопулярности мест.

### Задача 17. Центрированная карта

#### Входные данные.

Граф дорог Дистрикта (рис. 56)

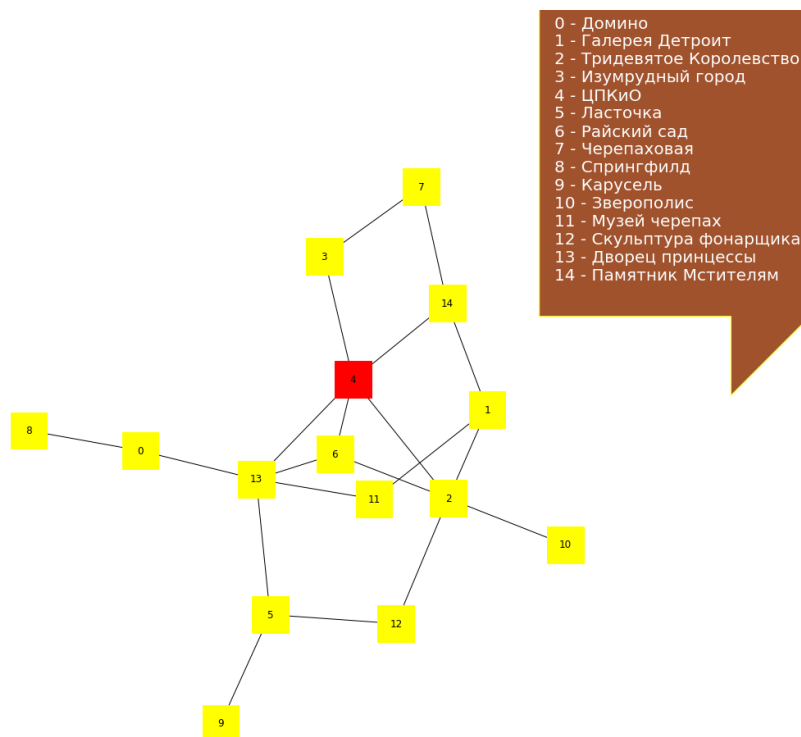


Рис. 56. Граф дороги Дистрикта

#### Условие задачи.

Центрированной точкой считается та, из которой сумма расстояний до остальных точек минимальна. Определите такую точку(и), исходя, из условной

карты дорог между точками района(графа) представленной ниже. При этом перекресток точкой не считается. Определите номер самой центрированной точки по району, если их несколько, то укажите одну(любую) из них.

### Решение

```
def Dijkstra(start, finish):
    global graph, size

    d = [float('inf') for i in range(size)]
    parent = [None for i in range(size)]
    visited = [False for i in range(size)]
    d[start] = 0

    for i in range(size):
        vertex = -1
        for j in range(size):
            if not visited[j] and (vertex == -1 or d[j] < d[vertex]):
                vertex = j
        if d[vertex] == float('inf'):
            break
        visited[vertex] = True

        for j in range(size):
            if graph[vertex][j] and (d[vertex] + 1) < d[j]:
                d[j] = d[vertex] + graph[vertex][j]
                parent[j] = vertex
    return d[finish]

graph = [
    [0,0,0,0,0,0,0,0,1,0,0,0,0,1,0],
    [0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,1],
    [0,1,0,0,1,0,1,0,0,0,1,0,1,0,0],
    [0,0,0,0,1,0,0,1,0,0,0,0,0,0,0],
    [0,0,1,1,0,0,1,0,0,0,0,0,0,0,1],
    [0,0,0,0,0,0,0,0,0,1,0,0,1,1,0],
    [0,0,1,0,1,0,0,0,0,0,0,0,0,0,1,0],
    [0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1],
    [1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
    [0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0],
    [0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
```

```
[0,1,0,0,0,0,0,0,0,0,0,0,1,0],  
[0,0,1,0,0,1,0,0,0,0,0,0,0,0],  
[1,0,0,0,1,1,1,0,0,0,0,1,0,0],  
[0,1,0,0,1,0,0,1,0,0,0,0,0,0]]
```

```
size = len(graph)
```

```
road_sum = []
```

```
for _from in range(size):
```

```
    d = []
```

```
    for _to in range(size):
```

```
        d.append(Dijkstra(_from, _to))
```

```
    road_sum.append(sum(d))
```

```
for i, value in enumerate(road_sum):
```

```
    print("For vertex # {}: {}".format(i, value))
```

```
For vertex # 0: 36  
For vertex # 1: 32  
For vertex # 2: 28  
For vertex # 3: 42  
For vertex # 4: 31  
For vertex # 5: 32  
For vertex # 6: 28  
For vertex # 7: 49  
For vertex # 8: 49  
For vertex # 9: 45  
For vertex # 10: 41  
For vertex # 11: 32  
For vertex # 12: 34  
For vertex # 13: 25  
For vertex # 14: 38
```

*Рис. 57. Результат работы программы поиска централизованной точки*

Из решения видно, что централизованной точкой является точка под номером 13.

**Ответ:** 13

*Задача 18. Непопулярность точки*

### Входные данные.

Посещаемость мест по дням недели для аналитика или файл с плотностью района, название проходного места (результаты решения задачи 17)

### Условия задачи.

Дополнительное пояснение по входным данным в интерактивной тетрадке и получению данных из нее:

- чтобы получить координаты мест необходимо после переменной указать `descret_coords` в квадратных скобках. Например, `data['descret_coords']`, где `data` – название вашей переменной с плотностью, выведет список всех координат;
- чтобы узнать посещаемость по часам по одному конкретному месту, необходимо обратиться к нему по его координатам. Например, если координаты места равны 1 и 2, то, чтобы получить посещаемость по каждому из 24 часов этой точки, обратитесь `data['distrib'][1][2]`, где `data` – название вашей переменной с плотностью.

Сравните данные посещаемости (по любому из файлов) центрированной точки (если их несколько, то выбранной вами в задаче 17) и самого посещаемого места. Сделайте соответствующие выводы.

Сделайте предположения почему центрированная точка может оказаться не самой посещаемой.

Опубликуйте свои выводы в эссе.

Вычислите у центрированной точки максимальное число посещений за день и укажите его в ответе к данному заданию.

### Решение.

```
find_vars(18)
coordinates = data_2_nsk["descret_coords"]
print(coordinates)
```



```
density_central_by_day = data_2_nsk['distrib'][4][18]
sum_central = 0
max_by_day = max(density_central_by_day)
for i in density_central_by_day:
    sum_central += i
density_most_visited_by_day = data_2_nsk['distrib'][3][8]
sum_most_visited = 0
for i in density_most_visited_by_day:
    sum_most_visited += i
print("Max visitors coeff: " + str(max_by_day))
print("Most visited average by day visitors coeff: " + str(sum_most_visited / 24))
print("Central average by day visitors coeff: " + str(sum_central / 24))
[[ 3  0]
 [ 0 14]
 [ 5 15]
 [ 0  1]
 [10  3]
 [ 9 15]
 [ 4  3]
 [19  2]
 [13 19]
 [17 15]
 [ 2 14]
 [11 15]
 [ 8 18]
 [ 6  2]
 [ 5  6]]
Max visitors coeff: 7.055910930742353
Most visited average by day visitors coeff: 3.4877742993205434
Central average by day visitors coeff: 3.3449060371788875
```

Рис. 58. Результат работы программы анализа непопулярности точек

**Ответ:**7

**Вывод**

Проанализировав посещаемость центрированной точки и сравнив ее с посещаемостью самой популярной точки, нами было замечено, что самая посещаемая точка не является центрированной.

Проанализировав полученные данные и результаты исследований из задач 1–17 можно сделать вывод, что центрированной точкой является тот

самый проблемный парк, с которым нам просит помочь мэр Дистрикта. Жители его не любят по некоторым причинам и именно поэтому не посещают.

### *Задача 19. Плотность района*

#### **Входные данные.**

Dataset-файл с массивами, информирующими о плотности района

#### **Условие задачи.**

Дополнительное пояснение по входным данным в интерактивной тетрадке и получению данных из нее:

- чтобы получить координаты мест необходимо после переменной указать `descret_coords` в квадратных скобках. Например, `data['descret_coords']`, где `data` - название вашей переменной с плотностью, выведет список всех координат;
- чтобы узнать посещаемость по часам по одному конкретному месту, необходимо обратиться к нему по его координатам. Например, если координаты места равны 1 и 2, то, чтобы получить посещаемость по каждому из 24 часов этой точки, нужно обратиться `data['distrib'][1][2]`, где `data` - название вашей переменной с плотностью.

Рассчитайте плотность Дистрикта города N по данным плотности района.

Проанализируйте данные по плотности Дистрикта. В каждой ячейке матрицы представленной в файле храниться информация о плотности в определенной точке города в определенное время (усредненно за год).

Укажите два числа через запятую (в итоге дробное число):

- усредненное значение плотности в районе в 14:00
- усредненное значение плотности для всего района

При этом укажите только целую часть, дробную часть получившихся значений отбросьте (например, 29,87 при отбрасывании дробной части станет 29).

Например: 23,28 где 23 - усредненное значение плотности в районе в 14:00, а 28 - значение плотности для всего район

### Решение.

```
import pickle
import pandas as pd

def get_mean_day_data(filename):
    data = None
    with open(filename, 'rb') as file:
        data = pickle.load(file)
    my_series = pd.Series(data)
    distrib = [0 for i in range(24)]
    day_amount = 0
    for days in my_series['distrib']:
        for day in days:
            distrib += day
            day_amount += 1

    distrib /= day_amount
    return distrib

region_density = get_mean_day_data("data_2_nsk.pickle")
region_mean = sum(region_density) / len(region_density)
print('.'.join([str(int(region_density[14])), str(int(region_mean))]))
```

Ответ  
5,3

*Задача 20. Поиск аналогичного по плотности района*

### Входные данные.

Dataset-файл с массивами, информирующими о плотности района, файлы с информацией о плотности трех других городов

### Условие задачи.

Дополнительное пояснение по входным данным в интерактивной тетрадке и получению данных из нее:

- чтобы получить координаты мест необходимо после переменной указать `descret_coords` в квадратных скобках. Например, `data['descret_coords']`, где `data` - название вашей переменной с плотностью, выведет список всех координат;
- чтобы узнать посещаемость по часам по одному конкретному месту, необходимо обратиться к нему по его координатам. Например, если координаты места равны 1 и 2, то, чтобы получить посещаемость по каждому из 24 часов этой точки, нужно обратиться `data['distrib']][1][2]`, где `data` - название вашей переменной с плотностью.

Проанализируйте данные о средней плотности из районов трех других городов и плотности вашего Дистрикта.

Вычислите усредненную плотность каждого района, отбросьте значение после двух знаков после запятой.

Сравните усредненную плотность своего района и районов других городов. Определите город аналог для вашего Дистрикта, это тот город из трех рассмотренных, у которого показатель плотности наиболее близок к значению плотности в вашем Дистрикте

Определите номер города-аналога плотности Дистрикта по номеру в имени файла (для задачи 25)

В ответ укажите:

- усредненное значение плотности вашего Дистрикта
- усредненное значение плотности города аналога

Формат ответа: усредненное значение вашего района-усредненное значение найденного города аналога. Через тире, без пробелов, при этом между

целой частью числа и дробной ставьте точку. Пробелов не должно быть в ответе.

Например: 23.23-24.45

### Решение.

```
import pickle
import pandas as pd
```

```
def get_mean_day_data(filename):
    data = None
    with open(filename, 'rb') as file:
        data = pickle.load(file)
    my_series = pd.Series(data)
    distrib = [0 for i in range(24)]
    day_amount = 0
    for days in my_series['distrib']:
        for day in days:
            distrib += day
            day_amount += 1
    distrib /= day_amount
    return distrib
```

```
region_density = get_mean_day_data("data_2_nsk.pickle")
region_mean = sum(region_density) / len(region_density)
min_diff = None
other_cities_mean = []
twin_mean = 0
for i in range(1, 4):
    city_density = get_mean_day_data('data_othrer_city{}.pickle'.format(i))
    other_cities_mean.append(sum(city_density) / len(city_density))
    if min_diff != None:
        if abs(region_mean - other_cities_mean[-1]) < min_diff:
            min_diff = abs(region_mean - other_cities_mean[-1])
            twin_mean = other_cities_mean[-1]
    else:
        min_diff = abs(region_mean - other_cities_mean[-1])
        twin_mean = other_cities_mean[-1]
```

```
s_region_mean = str(region_mean)
s_twin_mean = str(twin_mean)
```

```
print('-'.join([s_region_mean[:s_region_mean.find('.') + 3],  
s_twin_mean[:s_twin_mean.find('.') + 3]]))
```

**Ответ:** 3.40-3.40

*Задача 21. Посещаемость города-аналога*

### **Входные данные.**

Dataset-файл с массивами, информирующими о плотности района, столбчатая диаграмма по среднему значению загруженности мест в Дистрикте (задача 15, рис. 56)

### **Условие задачи.**

Задание программиста: построить столбчатую диаграмму по показателю о средней посещаемости мест Дистрикта.

Задание аналитика: взять данные из задачи 15, рис. 55, а данные по текущему шагу взять из результатов работы программиста команды по этому же заданию. Сравните полученные результаты по вашему Дистрикту из двух источников и сделайте выводы по точкам с низкой посещаемостью и максимальной. Оформите результаты сравнения в итоговом эссе.

### **Решение**

Столбчатая диаграмма в задании 15, рис. 31.

```
import matplotlib.pyplot as plt
```

```
def calculateVisitors():  
    places_average_density = []  
    for i in range(20):  
        total_sum = 0  
        for j in range(20):  
            sum = 0  
            for k in range(24):  
                sum += series["distrib"][i][j][k]  
            sum /= 24  
            total_sum += sum  
    total_sum /= 20
```

```
places_average_density.append(total_sum)
return places_average_density
```

```
with open('data_otrer_city2.pickle', 'rb') as f:
    data = pickle.load(f)
f.close()
series = pd.Series(data)
places = [str(i) for i in range(1, 21, 1)]
print(calculateVisitors())
plt.bar(places, calculateVisitors(), width=.5)
plt.gca().set_xticklabels(places)
plt.title("Средняя посещаемость объектов города-аналога за 20 дней",
          fontsize=22)
plt.ylim(0, max(calculateVisitors()))
plt.show()
```

Средняя посещаемость объектов города-аналога за 20 дней

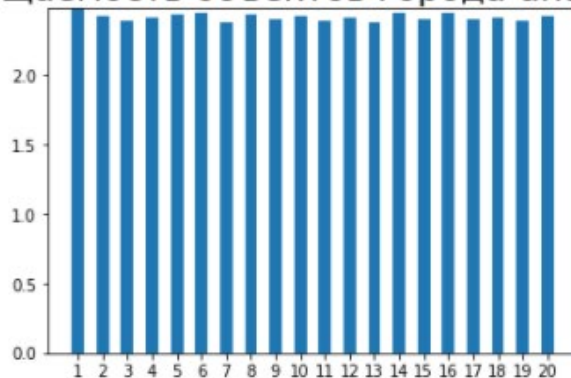


Рис. 59. Средняя посещаемость объектов города-аналога.

```
find_vars(21)
with open('data_2_nsk.pickle', 'rb') as f:
    data = pickle.load(f)
f.close()
series = pd.Series(data)
places = [str(i) for i in range(1, 21, 1)]
print(calculateVisitors())
plt.bar(places, list(calculateVisitors()), width=.5)
plt.gca().set_xticklabels(places, rotation=30, horizontalalignment='right')
plt.title("Средняя посещаемость объектов города Энск 2 Дистрикта за 20 дней",
          fontsize=22)
plt.ylim(0, max(calculateVisitors()))
plt.show()
```

Средняя посещаемость объектов города Энск 2 Дистрикта за 20 дней



Рис. 60. Средняя посещаемость объектов Дистрикта 2 г. Энск.

В целом данные по коэффициенту посещаемости за 20 дней не сильно отличаются, однако можно заметить, что в целом коэффициент посещаемости объектов в городе Энск выше на ~29%.

### Задача 22. Настроение города

#### Условие задачи.

Напишите функцию, которая по условию, указанному ниже, расшифровывает цветовой код улицы.

На вход подается цвет в формате RGB, например (55, 99, 20), а также массив значений цветового кода.

Массив значений цветового кода: colors = ["эlegantность", "сдержанность", "спокойствие", "стабильность", "эйфория", "нежность", "надежность", "великодушие", "экстравагантность", "превосходство", "величие", "бойкость", "благосостояние", "благородство", "плодовитость"]

Для определения настроения нужно вычислить некоторый "индекс" цвета. Его нахождение возможно с помощью следующего алгоритма:

Если  $r+g+b < 255$

то  $index = (r+g+b) // (255 // \text{длина массива colors})$  #здесь // - деление без остатка

Иначе Если  $r+g < 255$

то  $index = (r+g) // (255 // \text{длина массива colors})$

Иначе Если  $g+b < 255$

то  $index = (g+b) // (255 // \text{длина массива colors})$



Иначе

```
index = max(r, g, b) //(255//длина массива colors)
```

Полученный индекс будет являться индексом слова-настроения соответствующего цвету.

Для ответа запишите значение цвета одним словом. Например: величие.

**Решение.**

```
defcolor_code(rgb):
    r, g, b = rgb
    colors = ["элегантность", "сдержанность", "спокойствие", "стабильность",
"эйфория", "нежность", "надежность", "великодушие", "экстравагантность",
"превосходство", "величие", "бойкость", "благосостояние", "благородство",
"плодовитость"]
    index = 0

    if (r+g+b) <255:
        index = (r+g+b)//(255//len(colors))
    elif (r+g) <255:
        index = (r+g)//(255//len(colors))
    elif (g+b) <255:
        index = (g+b)//(255//len(colors))
    else: index = max(r,g,b)//(255//len(colors))

    return colors[index]

defget_each_color(rgb):
    rgb = rgb.replace('(', ')')
    rgb = rgb.replace(')', '(')
    rgb = rgb.replace(' ', '')
    returnlist(map(int, rgb.split(',')))
print(color_code(get_each_color(input())))
```

*Задача 23. Исследование климата*

**Входные данные.**

Dataset-файлы с климатическими условиями Дистрикта и климатическими условиями трех других городов

### Условие задачи.

Вычислите среднее значение температуры воздуха анализируемого вами района города N по данным, представленным в соответствующем файле.

По полученному среднему значению температуры Дистрикта найдите его "климатический" двойник (наиболее похожее значение) среди данных о районах других городов. Для нахождения двойника необходимо найти среднюю температуру для районов трех других городов. Округлите все результаты до двух знаков после запятой.

Укажите два числа: первым укажите среднее значение температуры (среднее климатическое района) вашего Дистрикта, вторым - его двойника. В качестве разделителя между целой и дробной частью числа используйте точку, а между значениями температуры используйте «;».

Примерответа: 23.23;4.45

### Решение.

```
import pandas as pd

our_mean = None
df = pd.read_csv('nsk_temp.csv')
our_mean = df['temp'].mean()

twin = 0
other_means = []
for i in range(1, 4):
    df = pd.read_csv('other_city{}.csv'.format(i))
    other_means.append(df['temp'].mean())

min_d = abs(our_mean - other_means[0])
twin = other_means[0]
for mean in other_means:
    if abs(our_mean - mean) < min_d:
        min_d = abs(our_mean - mean)
        twin = mean
print("{:.2f}; {:.2f}".format(our_mean, twin))
Ответ:28.67;27.57
```

### Задача 24. Исследование климата

#### Входные данные.

Среднее климатическое района: результаты, полученные при решении задачи 23.

#### Условие задачи.

Внесите в итоговую публикацию получившиеся ранее значение средней температуры воздуха исследуемого Дистрикта

**Ответ:** средняя температура воздуха в анализируемом Дистрикте составляет 23.23.

### Задача 25. Настроение города

#### Входные данные.

В качестве входных данных нужно взять ответ, полученный при решении 20 задачи, а именно: номер города аналога Дистрикта по посещаемости, работоспособный код по определению настроения города можно получить из решения задачи 22.

#### Условие задачи.

По номеру города аналога, возьмите из таблицы ниже координату точки. Номер соответствует номеру внутри файла анализируемых данных.

Таблица 3. Координаты некоторых точек городов-аналогов

Номер города аналога Дистрикта	Координаты для Google Карт
1	59°56'49.3"N 30°20'02.6"E
2	52°17'17.6"N 104°16'50.7"E
3	58°00'16.8"N 56°14'54.3"E

Отыщите эту точку на Google Картах, перейдите в режим панорамы. Сделайте принтскрин улицы. И с помощью сервиса определения главных цветов (см. задачу 5) определите три ключевых цвета улицы.

С помощью созданного в задаче 22 кода расшифруйте полученный цветовой код.

Запишите в эссе результаты расшифровки цветового кода улицы, добавьте принскрин улицы, название места с карты и сделайте вывод о том, имеет ли смысл скопировать этот цветовой код в вашем Дистрикте.

### Решение.

Номер города, усредненное значение плотности которого является аналогом Дистрикта 2, – 1, поэтому координаты улицы: 59°56'49.3"N 30°20'02.6"E

Название места: Центральный р-н, Санкт-Петербург, 191186.

Основные 3 цвета улицы и дистрикта представлены ниже, как и сама панорама улицы.

Вывод можно сделать следующий, дистрикту не хватает плодовитости.

```
def color_code(rgb):
```

```
    r, g, b = rgb
```

```
    colors = ["эlegantность", "сдержанность", "спокойствие", "стабильность",  
             "эйфория", "нежность", "надежность", "великодушие", "экстравагантность",  
             "превосходство", "величие", "бойкость", "благосостояние", "благородство",  
             "плодовитость"]
```

```
    index = 0
```

```
    if (r+g+b) < 255:
```

```
        index = (r+g+b)//(255//len(colors))
```

```
    elif (r+g) < 255:
```

```
        index = (r+g)//(255//len(colors))
```

```
    elif (g+b) < 255:
```

```
        index = (g+b)//(255//len(colors))
```

```
    else: index = max(r,g,b)//(255//len(colors))
```

```
    return colors[index]
```

```
def get_each_color(rgb):
    rgb = rgb.replace('(', ' ')
    rgb = rgb.replace(')', ' ')
    rgb = rgb.replace(',', ' ')
    return list(map(int, rgb.split(' ')))

region_rgb = ["(50,100,50)", "(150,150,150)", "(200,200,200)"]
print("Цветовые коды Дистрикта 2:")
for color in region_rgb:
    print(color_code(get_each_color(color)))

rgb = ["(100,100,100)", "(150,150,150)", "(100,150,200)"]

print("\nЦветовые коды улицы:")
for color in rgb:
    print(color_code(get_each_color(color)))
```

```
Цветовые коды Дистрикта 2:
бойкость
экстравагантность
бойкость

Цветовые коды улицы:
бойкость
экстравагантность
плодовитость
```

Рис. 61. Текстовый результат работы программы

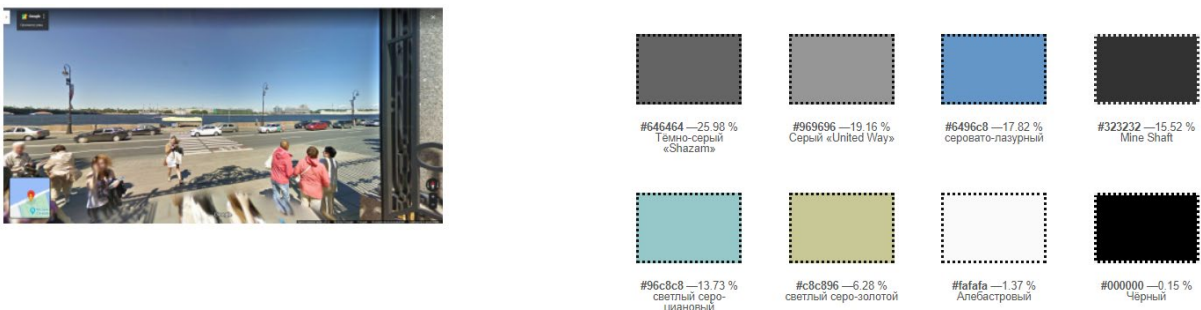


Рис. 62. Снимок экрана улицы и его цветовой анализ

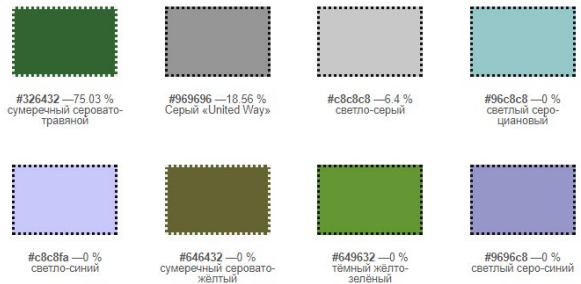


Рис. 63. Фотография дистрикта и её цветовой анализ

## Вывод

Проанализировав цветовую палитру города аналога (рис. 61, 62), мы пришли к выводу, что городу Энску не стоит копировать полученный цветовой код, ведь в основном, в палитре города (рис. 63) аналога преобладает серый и его оттенки.

## Задача 26. Центрированная карта

### Входные данные.

Карта Дистрикта с геометками (см. задачу 10), граф дорог Дистрикта (рис. 56)

### Условие задачи.

Соотнесите точки из графа ниже с точками на графической карте с геометками вашего района, получившейся на шаге 18. И отобразите все дороги между инфраструктурными точками. Разместите новую карту района в своем эссе (тетрадь Google Collab).

### Решение.

На карте ниже (рис. 64) представлены места и соединяющие их дороги на основе представленного в условии задачи графа (рис. 56).

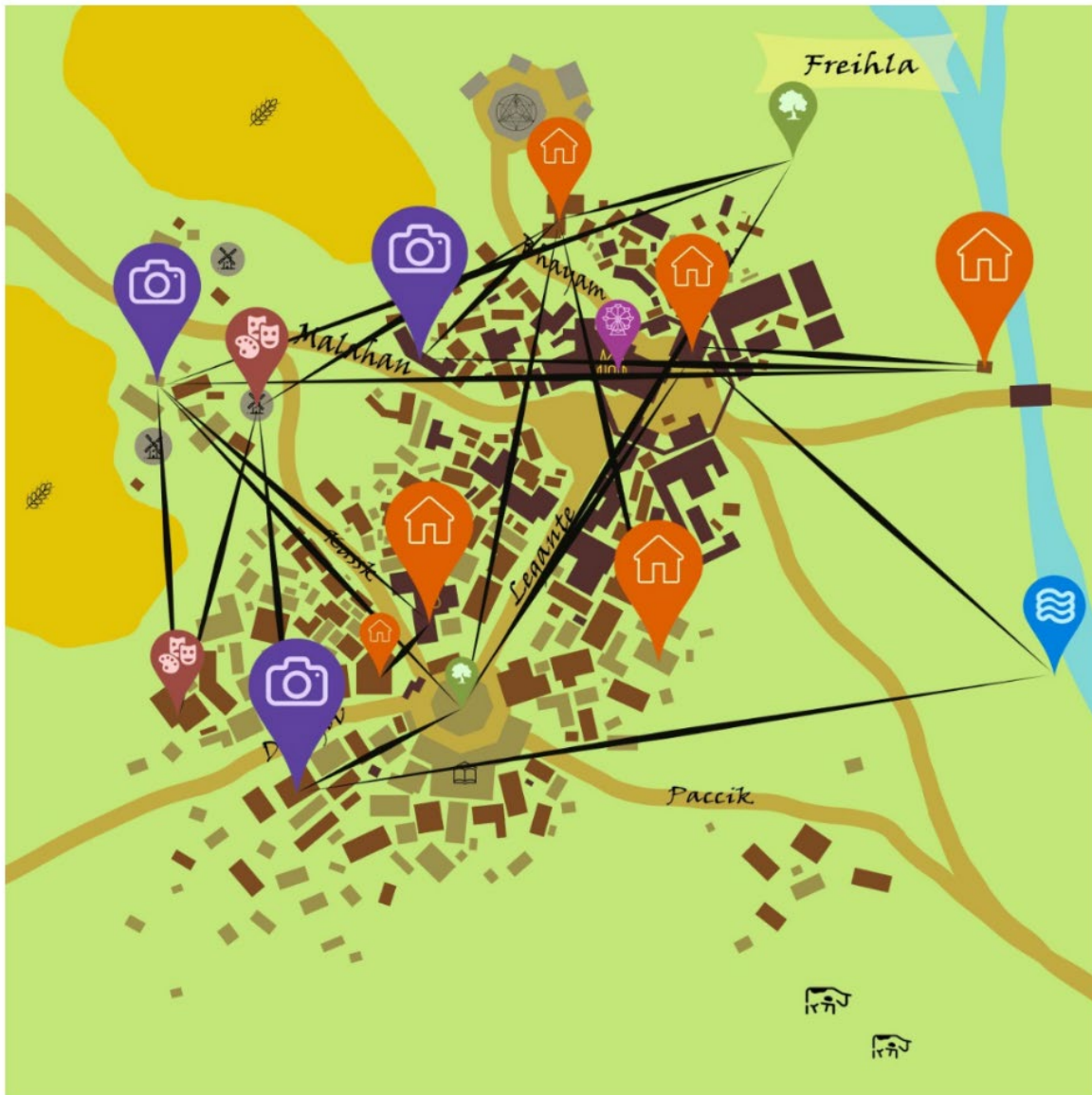


Рис. 64. Места Дистрикта и соединяющие их дороги

### Задача 27. Добавление дорог

#### Входные данные.

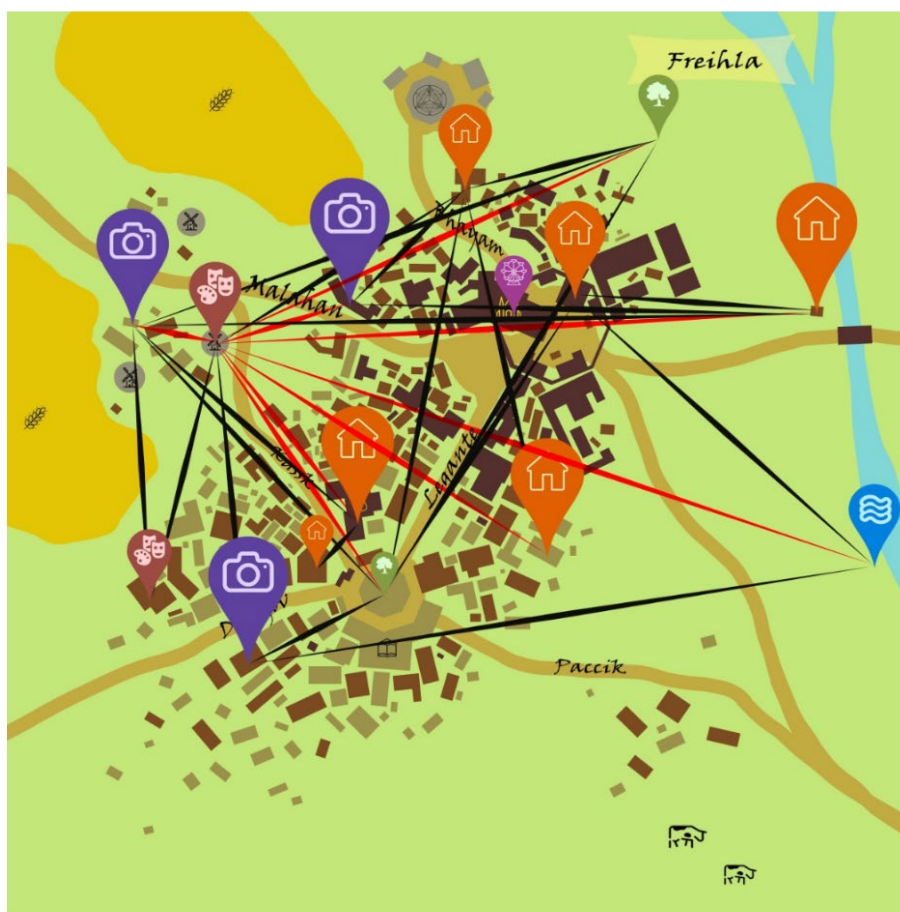
Самая посещаемая точка, самая центрированная точка, граф дорог в Дистрикте согласно условиям и решениям предыдущих задач.

#### Условие задачи.

Модифицируйте карту Дистрикта таким образом, чтобы самая посещаемая точка стала самой центрированной. Для этого вы можете добавить новые дороги между пунктами.

**Решение.**

На рис. 65 все дороги ведут в центрированную точку. Черным цветом помечены новые дороги. Красным – имеющиеся дороги.



*Рис. 65. Дополнительные дороги на карте города*

*Задание 28. Сопоставление негативных и позитивных отзывов*

**Входные данные.**

Отзывы о значимых точках района (задача 4), результаты анализа отзывов (задача 4)

**Условия задачи**



Семантический анализ подразумевает распределение отзывов пользователей по настроению: негативному и позитивному.

Отзывы имеют оценку пользователей. Считается, что рейтинг от 3 и ниже определяет отзыв как негативный, а позитивные имеют оценку 4 или 5.

Проанализируйте количество негативных и позитивных отзывов в Дистрикте.

Составьте диаграмму-пирог по анализу количества негативных и позитивных отзывов и разместите ее в своем эссе (тетрадь Google Collab).

По итогу анализа укажите количество негативных и количество позитивных отзывов.

Форма ответа: количество негативных отзывов, количество позитивных отзывов через запятую (В итоге дробное число).

Например:

13,32 где 13 - количество негативных отзывов, а 32 - положительных

26,19 где 26 - количество негативных отзывов, а 19 - положительных

### Решение.

```
user_ratings = [r.text for r in soup.find_all("span", {"class": "rating"})]
negative = 0
positive = 0
user_ratings_dict = {}
for i in user_ratings:
    if int(i) > 3:
        positive += 1
    else:
        negative += 1
if not in user_ratings_dict.keys():
    user_ratings_dict.update({i: 1})
else:
    user_ratings_dict.update({i: user_ratings_dict.get(i) + 1})
print(str(negative) + "," + str(positive))
labels = user_ratings_dict.keys()
values = user_ratings_dict.values()
colors = ['gold', 'yellowgreen', 'lightcoral', 'lightskyblue', 'white']
```

```
plt.pie(values, labels=labels, colors=colors,  
autopct='%1.1f%%', shadow=True, startangle=140)  
plt.axis('equal')  
plt.show()
```

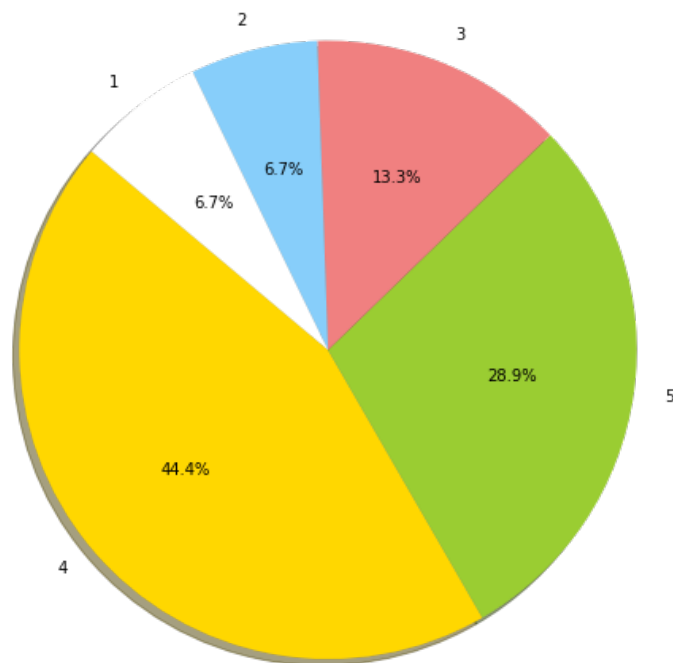


Рис. 66. Диаграмма распределения оценок

Отзывов с оценкой:

1 - 6.7%

2 - 6.7%

3 - 13.3%

4 - 44.4%

5 - 28.9%

**Ответ:** 12,33

*Задание 29. Улучшение системы дорог*

**Входные данные.**

Список дорог, которые нужно добавить к имеющимся (задача 27), карта Дистрикта с геометками и дорогами (рис. 65)

**Условия задачи.**

Нанесите на созданную вами карту с геометками и дорогами дополнительные дороги, которые спроектировали на шаге 27. Выделите их цветом или формой.

**Решение**

Решение показано на рис. 65. Черным цветом помечены существующие дороги.

*Задание 30. Оформление вычислительного эссе***Входные данные.**

Графический материал, полученный в ходе работы всеми участниками команды.

**Условия задачи.**

В результате выполнения командой заданий появилось несколько графических объектов. Добавьте на них подписи, визуализируйте ответ при помощи выделения нужно столбца или сектора на них, установите на графике метки

Данные, появляющиеся в ходе работы:

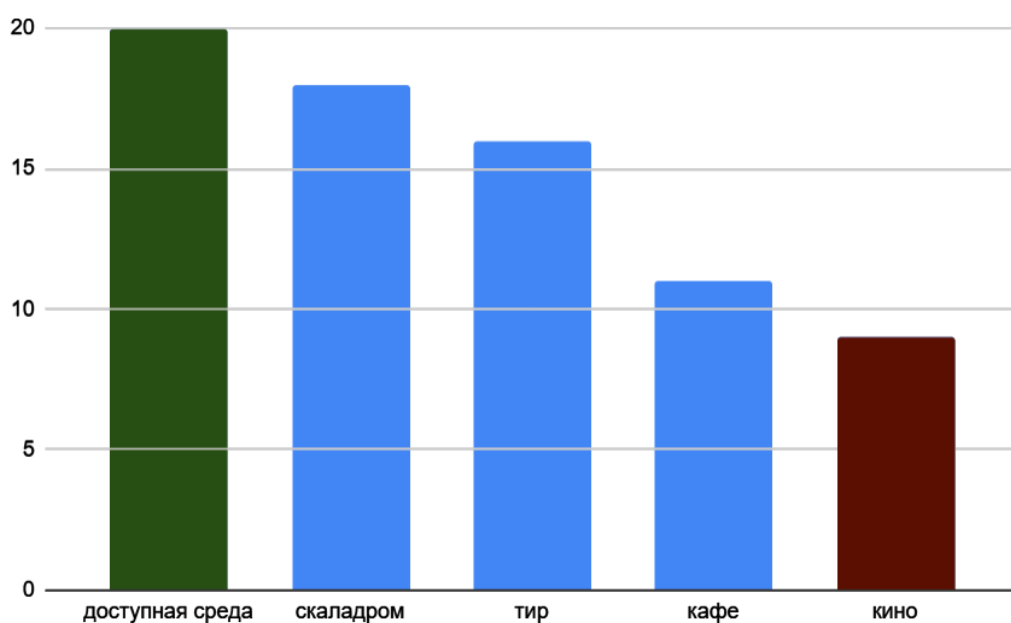
- задача 5: диаграмма-пирог с результатом анализа насаждений в Дистрикте;
- задача 7: график интересов Дистрикта;
- задача 12: облако часто упоминаемых слов;
- задача 13: облако слов позитивного настроения, облако слов негативного настроения;
- задача 15: столбчатая диаграмма по среднему значению загруженности мест в Дистрикте;
- задача 21: график плотности мест Дистрикта;

- задача 28: диаграмма-пирог по анализу количества негативных и позитивных отзывов.

Примечание. У вас могут появиться дополнительные графики, которые вы захотите вставить в заключительное эссе, их тоже необходимо отредактировать.

### Решение (примеры)

#### 1. Изменить график интересов Дистрикта (рис. 67)



*Рис. 67. Отредактированный график интересов Дистрикта*

#### 2. График по среднему значению загруженности мест в Дистрикте (рис. 68)

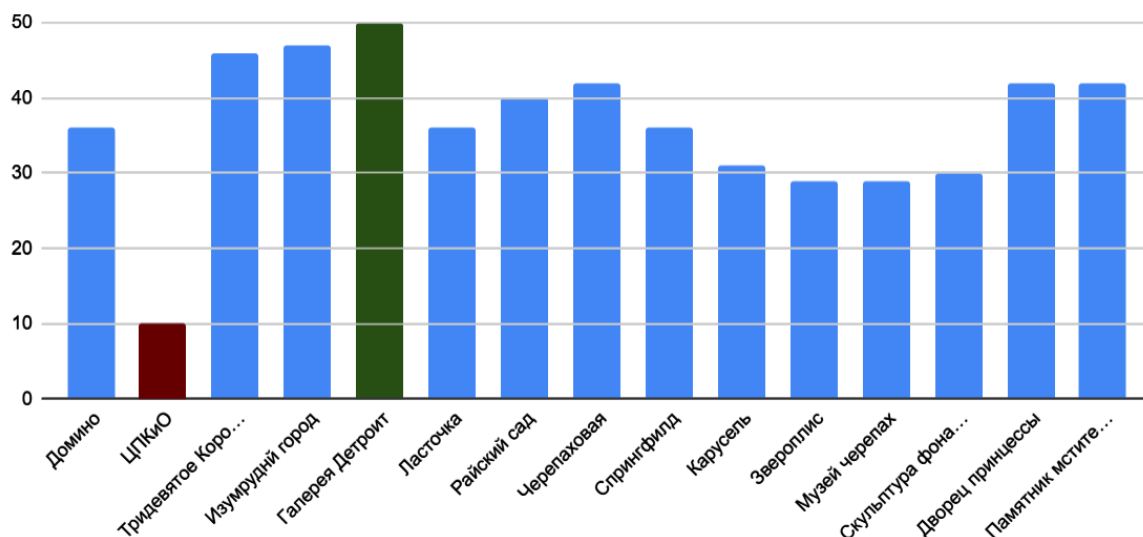


Рис. 68. Отредактированный график загруженности мест Дистрикта

### Задание 31. Новые объекты городской инфраструктуры

#### Входные данные.

Все материалы полученные в процессе исследования Дистрикта.

#### Условия задачи.

Исходя из полученных данных в ходе анализа по Дистрикту города N (погода, плотность, его текущая инфраструктура и т. п.), выбрать места куда лучше всего установить новые объекты городской инфраструктуры. Обосновать потребность в новых объектах не только логически, но и результатами анализа данных по району. Показать, как может меняться настроение города в зависимости от места и положения установленного вами объекта.

Нанести новые объекты на итоговую карту Дистрикта.

Оформить все выводы в эссе.

Подготовиться к защите и демонстрации своего вычислительного эссе: итоговой тетради в Google Collab. На демонстрацию тетради вам выделяется 5

минут. Рекомендованное количество докладчиков: два (например, журналист и аналитик).

### **Решение.**

Так как жителям не нравился парк ЦПКиО, в частности из-за того, что он был построен давным-давно, что понятно из статьи мэра; а также исходя из интересов жителей к скалодрому, тир и кино, можно предложить добавить торгово-развлекательный комплекс возле ЦПКиО, который за счет того, что у парка удачное расположение, тоже станет центрированным, а также, поднимет популярность посещения парка, если в нем расположить тир. Внутри ТРЦ можно обустроить кафе и скалодром. После активного отдыха, просмотра кино и шоппинга, люди смогут отдохнуть в парке. Таким образом сразу удовлетворятся несколько потребностей жителей Дистрикта.

Для удовлетворения потребностей жителей в доступной среде предлагаем создавать ТРЦ сразу с учетом данного фактора и реконструировать парк так, чтобы ветки не мешали ни ходьбе, ни проезду, и сделать в парке новые ровные дорожки.

Можно предложить добавить еще один парк, в этот раз уже за пределами города: летом - замечательное место для прогулок, зимой - горнолыжная база. Так как среди самых популярных запросов пользователей значилась комфортная среда и кафе, можно расположить кафе рядом со всеми значимыми объектами инфраструктуры, будь то жилой квартал или развлекательный центр. Также среди цветовых кодов была “бойкость”, в связи с чем можно принять решение добавить спорткомплекс в относительной доступности, который включает стадион, бассейн, скалодром, тир и т.д.

### *Задание 33. Манипуляторы*

## Условия задачи

Для того, чтобы территория процветала, и среда обитания населения города была комфортной, а люди счастливыми, необходимо наделить район объектами в соответствии с интересами населения. Каждый объект имеет оценку, указывающую на степень одобрения его жителями района – "баллы счастья" (можно посмотреть условие задачи 9 первой попытки).

Администрация может разместить в районе:

- пункты сбора раздельного мусора (waste) - 12 баллов за один пункт;
- открыть общепиты/фудкорты (cafe) - 12 баллов за одно предприятие;
- настроить вай-фай на площадках (wifi) - 12 баллов за одну площадку;
- элитные магазины (shop) - 17 баллов за 1 магазин;
- сделать парковки (parking) - 16 баллов за одну парковку;
- открыть почтовые отделения (postoffice) - 21 балл за 1 офис;
- открыть развлекательные пункты тира (tir) - 16 баллов за 1 пункт;
- сделать площадки скалодрома (scaladrom) - 18 баллов за 1 скалодром;
- открыть поликлиники (polyclinic) - 15 баллов за 1 поликлинику.

Кроме того, умные решения требуют финансовых затрат, которые у нас ограничены. Городом для администрации района выделено — 282 000 условных денежных единиц (уде), при этом:

- на пункт мусора затрачивается 7000 уде;
- на предприятие общественного питания - 5000 уде;
- на wi-fi точку - 15000 уде;
- магазин - 15000 уде;
- парковка - 8000 уде;
- почтовое отделение - 7000 уде;
- тир – 1500 уде;
- скалодром – 5000 уде;

- поликлиника – 12000 уде.

Определите оптимальное количество умных решений достаточных для обеспечения комфортной среды обитания района и укладывающихся в бюджет.

Параметр необходимого уровня комфорта устанавливается таким образом, чтобы его значение было не ниже максимального показателя в других районах города (жители которых довольны обустройством и жизнеобеспечением территории): 550, 450, 500, 480.

Для ввода ответа к этой задаче - введите в поле ввода на сайте (<https://yupest.github.io/nti/sliders/check2.html>) количество каждого объектов через пробел. Если значения являются решением задачи, то ниже поля ввода появится код. Скопируйте код и вставьте его в качестве ответа к этой задаче

### Решение.

```
fromipywidgetsimport Dropdown, Checkbox, HBox, VBox, Text, Button
fromIPython.displayimport display, clear_output
```

```
defcounts_of_points(waste = 5, cafe = 5, wifi = 5, shop = 5, parking = 5, post_office
= 5, tir = 5, scaladrom=5, polyclinic=5):
```

```
mwaste = waste*7000
mcafe = cafe*5000
mwifi = wifi*15000
```

```
mshop = shop*15000
mparking = parking*8000
mpost_office = post_office*7000
mtir = tir*1500
mscaladrom = scaladrom*5000
mpolyclinic = polyclinic*12000
```

```
spends = [mwaste, mcafe, mwifi, mshop, mparking, mpost_office, mtir,
mscaladrom, mpolyclinic]
```

```
maxspend = 282000
```



```
title_1 = "Текущиетраты %d!"%sum(spends) ifsum(spends)
<maxspendelse"Текущиетраты %d! Превышаютлимит %s"% (sum(spends) ,
str(maxspend))
```

```
max_happy = waste*12 + wifi*12 + cafe*12 + shop*17 + parking*16 +
post_office*21 + tir*16 + scaladrom*18 + polyclinic*15
```

```
other_centers = [550, 450, 500, 480, max_happy ]
```

```
explode = explode = (0, 0, 0, 0, 0.1)
```

```
title_2 = "Счастьяслишкоммало!"ifmax(other_centers[:-1]) >other_centers[-1]
else"Счастьяскольконодо!"
```

```
plt.figure(figsize = (10, 4))
```

```
plt.subplot(1,2,1)
```

```
plt.pie(spends,labels=['waste', 'cafe', 'wifi', 'shop', 'parking', 'post_office', 'tir',
'scaladrom', 'polyclinic'], autopct='%1.1f%%',
```

```
shadow=True, startangle=90)
```

```
plt.title(title_1)
```

```
plt.subplot(1,2,2)
```

```
plt.pie(other_centers,labels=other_centers, explode = explode,
autopct='%1.1f%%',
```

```
shadow=True, startangle=90)
```

```
plt.title(title_2)
```

```
plt.tight_layout()
```

```
plt.show()
```

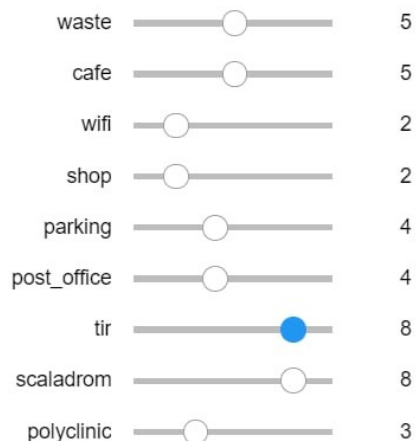
```
importmatplotlib.pyplotasplt
```

```
fromipywidgetsimport interact, interactive, fixed, interact_manual
```

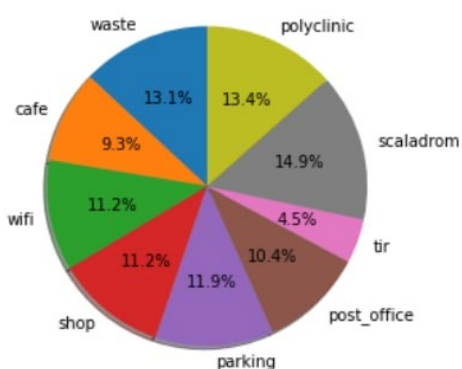
```
fromIPython.displayimport display
```

```
w = interactive(counts_of_points, shop = (0,10), parking = (0,10), post_office =
(0,10), tir = (0,10), polyclinic = (0,10), scaladrom=(0,10), waste = (0,10), cafe =
(0,10), wifi = (0,10))
```

```
display(w)
```



Текущие траты 268000!



Счастья сколько надо!

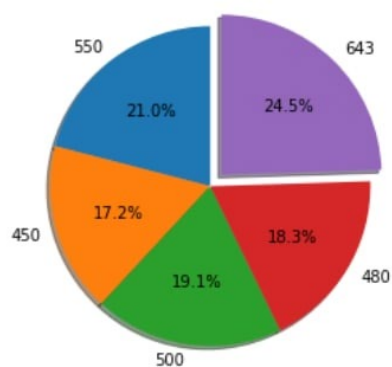


Рис. 69. Результат работы программы построения диаграмм с манипуляторами.

Исходя из интересов жителей Дистрикта было решено, что нужно больше тиров и скалодромов.

Оптимальный расход определяется так:

- 2 waste - 24
- 10 cafe - 120
- 2 wifi - 24
- 3 shop - 51
- 3 parking - 48
- 3 post\_office - 61
- 14 tir - 224
- 13 scalodrom - 234
- 1 hosptal - 15

По сумме вышло ровно 282 уде. Индекс счастья - 801

**Ответ:** 2 10 2 3 3 3 14 13 1





Во время решения задач командного этапа ребятам предстоит самим определить соответствующий направленности задачи навык. Пример такого сопоставления для задач отборочного тура приведен в таблице ниже.

*Сопоставление типов задач и методов их решения*

№	Основное метод решение	Альтернативный подход к решению	Тип задачи	Пример задачи из отборочного тура
1	Создание тетрадки Google Colab	Создание публикации в Power Poin (крайне не желательный вариант, но возможный вариант)	Осуществление вычислений, оформление публикации исследования	Как общая задача, рассмотрен в пробных задачах финала и видео-инструкции к ним.
2	Парсинг данных с сайта	Найти и скопировать нужные данные с сайта в ручную	Задача на нахождение тегов определенных данных	<b>Задача 1. Сбор данных</b> См. Приложение 2 к данному документу.
3	Вычисления в тетрадке на языке (Python)	Расчеты в Excel	Задачи на нахождение плотности (населения, застройки, транспортных развязок и т.п.), определение климатической зоны, калькулятор перевода для единиц измерения.	<b>Задача 2. Плотность населения</b>
4	Построение облака слов (Python)	Сервис в интернете: <a href="https://wordsccloud.pythonanywhere.com">https://wordsccloud.pythonanywhere.com</a>	Задачи на обработку текстов отзывов и выделение наиболее часто употребляемых в них слов. Построение облака слов.	<b>Задача 3. Анализ отзывов</b>
5	Построение линейчатых графиков Python	Графики и анализ данных в Excel	Задачи на построение рейтинга, анализ погодных условий, расписания работы	<b>Задача 4. Формирование рейтинга</b>

			(временных интервалов) и т.п.	
6	Построение столбчатых графиков (Python)	Графики и анализ данных в Excel	Задачи на определение рейтингов, наибольших и наименьших значений различных категорий, установления лидеров и т.п.	<b>Задача 5. Посещаемость развлекательных центров</b>
7	Построение круговых диаграмм (Python)	Графики и анализ данных в Excel	Задачи на определение секторов: мнения аудитории, погодные условия, распределение средств на различные нужды и т.п.	<b>Задача 6. Анализ целевой аудитории</b>
8	Построение интерактивных диаграмм с манипуляторами и (Python)	Решить аналитически	Задачи с параметрами: изменение параметра влияет на результат, оптимальный результат подбирается регулированием различных параметров через интерфейс манипуляторов	<b>Задача 7. Проектирование парковых зон</b>
9	Задачи на работу с массивами и списками (Python)	Анализ данных в Excel	Задачи на анализ данных в массивах. Как правило, поиск мест (точек) на координатной плоскости, отвечающих определенным характеристикам.	<b>Задача 8. Замечательные места</b>
10	NumPy: сложные задачи (Python)	Нет аналога	Задачи на анализ пространственно временных данных	<b>Задача 9. Как найти проходное место?</b>
11	Работа с картами библиотека	Нет аналога	Задача разметки карты: указания на ней дороги, типов объектов городской инфраструктуры при	Задача будет рассмотрена во время подготовки к финалу

			помощи специальных меток	
12	Обработка графиков в графическом редакторе Paint.	Любой графический редактор, в котором удобно работать	В задачу входит обработка построенных программистом графиков иллюстрации	Задача будет рассмотрена во время подготовки к финалу

	Журналист данных
	Дизайнер
	Аналитик
	Программист