Санкт-Петербургский Национальный Исследовательский Университет Информационных Технологий, Механики и Оптики

Факультет Инфокоммуникационных Технологий

**Домашняя работа №3**

**Вариант № 4**

Выполнили:

Бакланова А.Г,

Крылов Д.С,

Улитина М.С

Проверил Мусаев А.А.

Санкт-Петербург, 2023

Оглавление

[Оглавление 2](#_Toc130762759)

[Введение. 3](#_Toc130762760)

[Алгоритмы выполнения заданий. 4](#_Toc130762761)

[Задание 1. 4](#_Toc130762762)

[Задание 2. 5](#_Toc130762763)

[Задание 3. 7](#_Toc130762764)

[Задание 4. 7](#_Toc130762765)

[Заключение. 9](#_Toc130762766)

# 

# Введение.

Цель лабораторной работы – выполнение задания с помощью Python по теме «Жадные алгоритмы». Вариант работы – 4.

# 

# Алгоритмы выполнения заданий.

## Задание 1.

В первом задании нам необходимо реализовать программу, которая считает наименьшую комбинацию из заданных монет для выдачи сдачи.

На рисунке 1 приведен код программы для первого задания. Задаем переменную k = 0, которая нужна для подсчета общего количества монет. После создается список, состоящий из списков, где хранится информация о номинале монеты и количестве этих монет, и сортируем его. Переменным для подсчета количества каждой монеты присваиваем значения «0». В 17 строке проверяем, что сумма сдачи не превышает суммы имеющихся монет. После берем как можно больше самых крупных монет, если этого сделать не можем, то берем номинал меньше и так до конца.

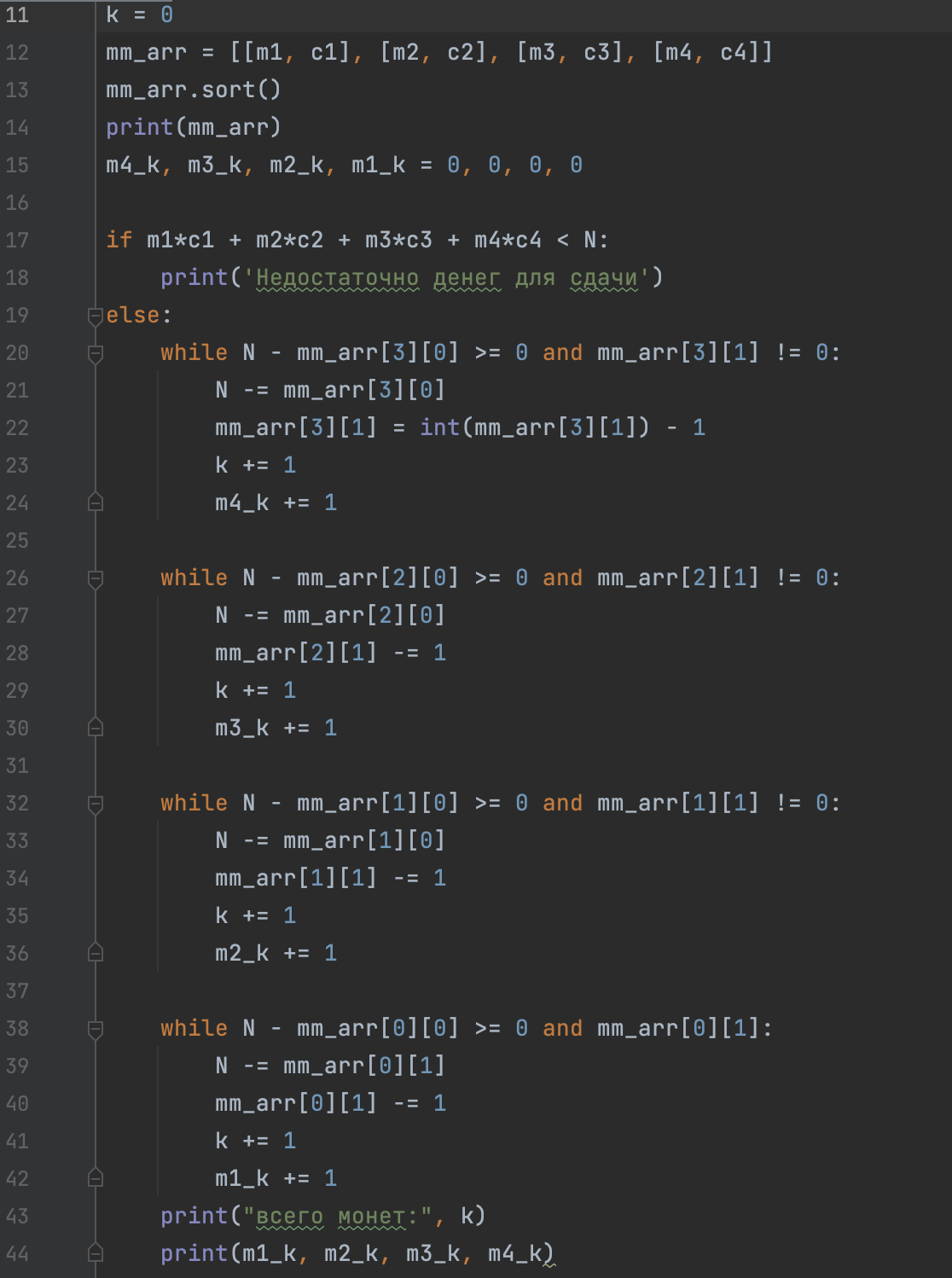


Рисунок 1 – Программа для сдачи.

На рисунке 2 приведен вывод программы.

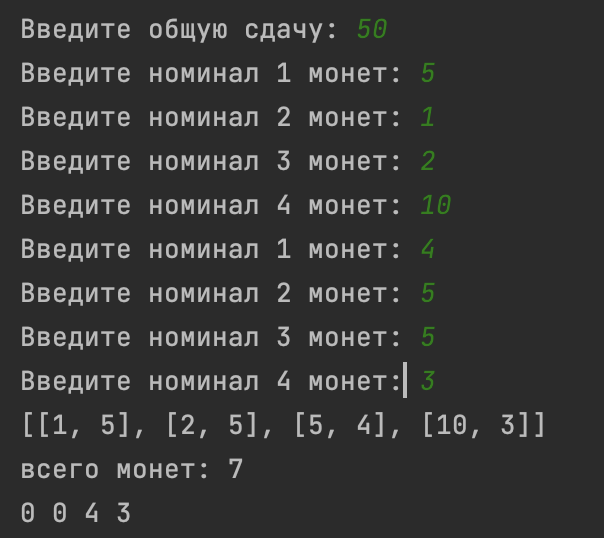


Рисунок 2 – Результат работы программы.

## Задание 2.

Во втором заданий нужно определить, как вору вынести наибольшую сумму, исходя из количества заходов и доступного количества килограмм за один заход.

На рисунке 3 приведен код реализации. Создаем функцию, общая сумма равна 0. Сортируем экспонаты по стоимости за килограмм. Проверяем какие экспонаты мы не можем украсть. После чего пробуем забирать экспонаты пока есть возможность. С 13 строки забираем экспонаты, которые можем забрать целиком. В 20-21 проверяем, что мы можем унести этот экспонат. Иначе берем самый дорогой (23-29 строки). Возвращаем значения общей суммы, украденных экспонатов.



Рисунок 3 -Программа для вора.

На рисунке 4 приведен список экспонатов (вес, цена) и строки для ввода и вывода.

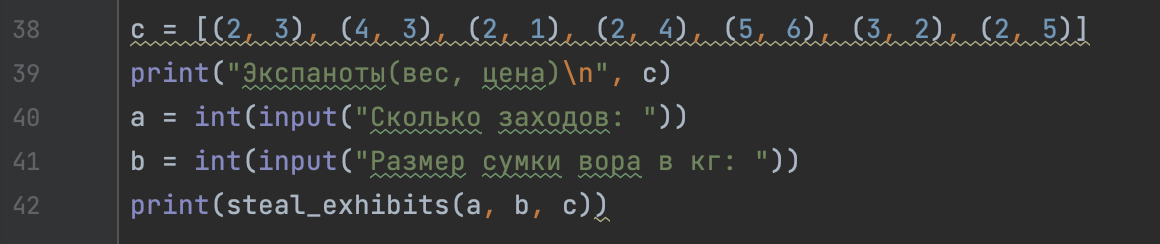


Рисунок 4 – Вывод в программе.

На рисунке 5 результат работы программы.

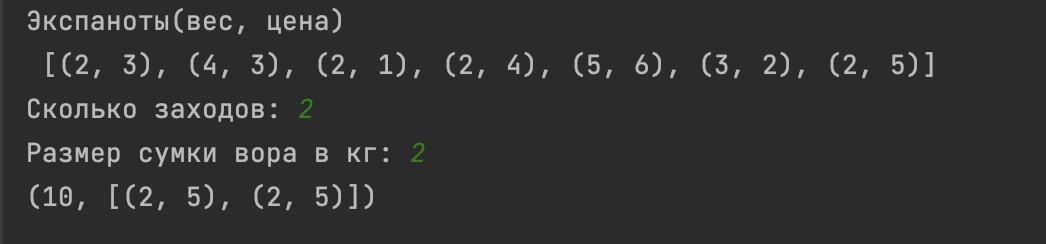


Рисунок 5 – Результат работы программы.

## Задание 3.

Необходимо сделать выводы о «жадных» алгоритмах и ответить на вопрос: «Всегда ли наилучшие решения на каждом шаге приводят к наилучшему конечному результату?».

Мы считаем, что жадные алгоритмы могут быть полезны, например, в задании 1 действительно будет искать наилучшее выражение, так как чем больше монета, тем меньше их потребуется. Но, в случае задания 2, не учитываются многие комбинации экспонатов, и возможна ситуация, когда комбинация дешевых экспонатов в большем количестве будет выгоднее.

Итак, не всегда наилучшие решения на каждом шаге приводят к наилучшему конечному результату.

## Задание 4.

Мы реализовали алгоритм Дейкстры на основе городов России (4 вариант). На рисунке 6 приведен алгоритм работы программы. В начале определяется функция dijkstra, которая принимает на вход граф и начальную вершину. Создаются словарь distances, в котором для каждой вершины графа устанавливается бесконечное расстояние до нее, за исключением начальной вершины, до которой расстояние устанавливается равным 0. Затем создается очередь с приоритетом queue, в которую добавляется кортеж из расстояния до вершины и самой вершины.

Далее начинается основной цикл, который выполняется до тех пор, пока очередь не станет пустой. Из очереди извлекается вершина с наименьшим расстоянием, и если это расстояние больше, чем уже известное минимальное расстояние до этой вершины, то цикл продолжается. В противном случае происходит обновление расстояния до всех соседних вершин текущей вершины, если новое расстояние меньше уже известного.

В конце функция возвращает словарь distances, в котором для каждой вершины указано кратчайшее расстояние до нее от начальной вершины.

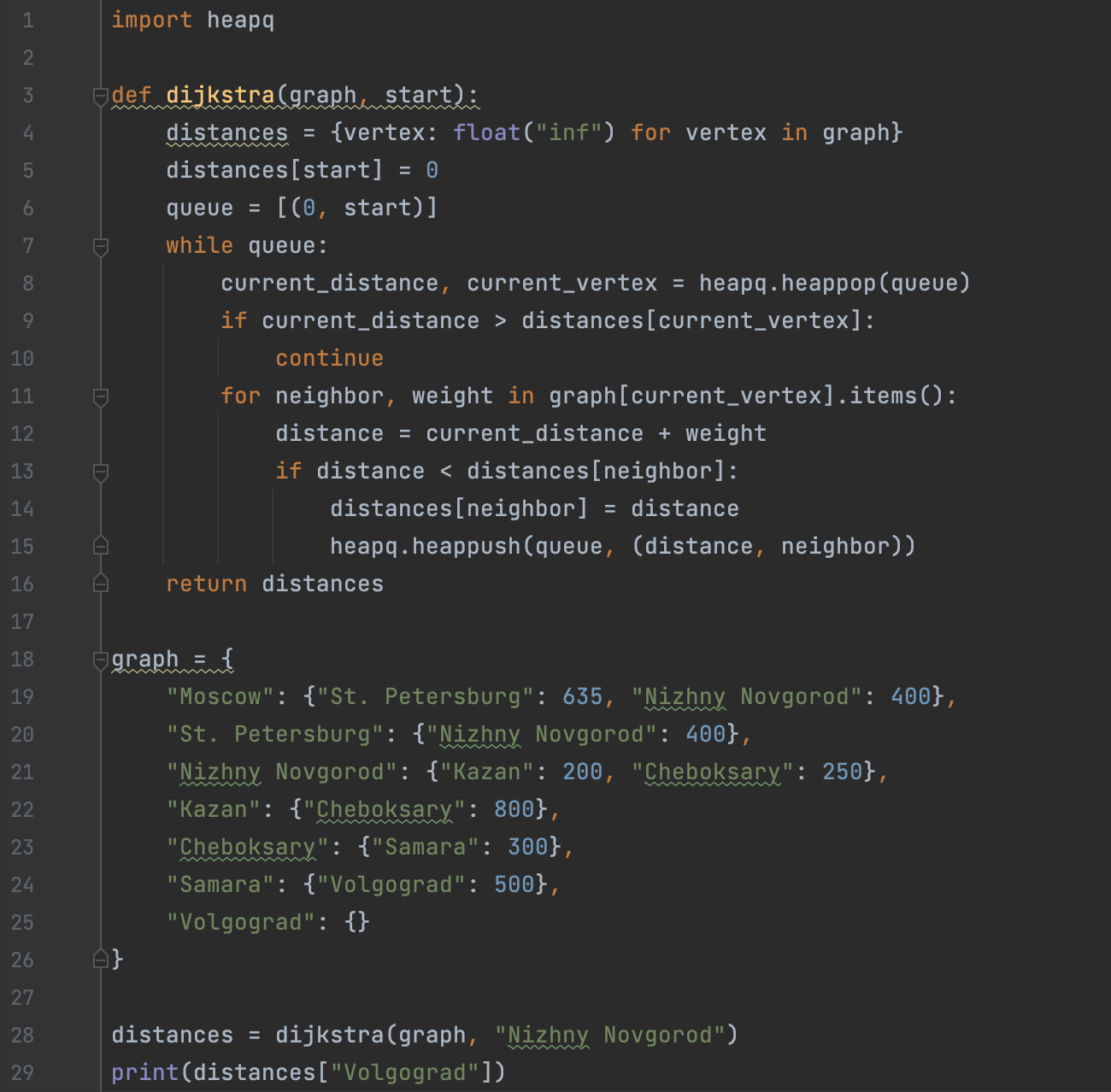


Рисунок 6 – Задание 4.

Результат работы программы приведен на рисунке 7.



Рисунок 7 – Результат работы программы.

# Заключение.

Задание выполнено, таким образом, можно считать, что цель домашней работы выполнена. Задание выполнено с учетом варианта. Вариант работы: 4. Изучены «жадные алгоритмы». Программы находятся по ссылке: <https://github.com/Anasstasssia/hw3-3>