

Лабораторная работа 4.

Функции

Требования к выполнению заданий

1. На оценку 3 балла:
 - а) реализовать программу в соответствии с заданием.
2. На оценку 4 балла:
 - а) добавить проверку входных аргументов и вывод сообщений об ошибках (используя исключения);
3. На оценку 5 баллов:
 - а) реализовать интерфейс командной строки;
 - б) оценить полученный алгоритм в нотации « O » большоеПлюсом будет наличие документации полученной программы.

Задания

Вариант 1

Напишите функцию, реализующую решето Эратосфена.

Входные значения: верхняя граница интервала $[0, N]$.

Выходные значения: список простых чисел.

Вариант 2

Написать функцию, реализующую алгоритм бинарного поиска.

Входные значения: отсортированный список длиной n , значение, которое необходимо найти в списке.

Выходные значения: индекс элемента в списке, если таковой найден или None если элементов нет.

Вариант 3

Написать функцию, реализующую алгоритм сортировки выбором.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 4

Написать функцию, реализующую алгоритм поиска в ширину.

Входные значения: граф, в котором производится поиск, элемент который необходимо найти.

Выходные значения: True если элемент есть в графе, иначе False.

Вариант 5

Напишите функцию, реализующую решение задачи покрытия множества с помощью жадных алгоритмов.

Входные значения: словарь, ключи которого наименование радиостанций, а значения – множества регионов, которые покрывают радиостанции; множество всех регионов, которые необходимо покрыть.

Выходные значения: минимальное количество станций, множество которых покрывает все регионы.

Жадный алгоритм:

а) выбрать станцию, покрывающую наибольшее количество регионов, еще не входящих в покрытие. Если станция будет покрывать некоторые регионы, уже входящие в покрытие, это нормально.

б) повторять, пока остаются непокрытые элементы множества.

Вариант 6

Напишите функцию, реализующую алгоритм поиска Боуэра-Мура.

Входные значения: строка, в которой необходимо произвести поиск; множество слов которые необходимо найти.

Выходные значения: словарь, ключи которого найденные слова, а значения – кортежи индексами начала и конца слова в исходной строке.

Вариант 7

Напишите функцию, реализующую поиск счастливые простые числа.

Входные значения: верхняя граница интервала $[0, N]$.

Выходные значения: список счастливых простых чисел.

Вариант 8

Написать функцию, реализующую алгоритм Дейкстры.

Входные значения: взвешенный граф, стартовый узел, целевой узел.

Выходные значения: кортеж, состоящий из дистанции между начальным и целевым узлом и путь из начального в целевой узел в виде словаря.

Вариант 9

Написать функцию, реализующую алгоритм сортировки Шелла.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 10

Написать функцию, реализующую алгоритм Прима.

Входные значения: взвешенный граф в виде хеш-таблицы (словаря).

Выходные значения: минимальное оставное дерево в виде словаря.

Вариант 11

Написать функцию, реализующую алгоритм поиска в глубину.

Входные значения: граф (в виде словаря), в котором производится поиск, элемент который необходимо найти.

Выходные значения: True если элемент есть в графе, иначе False.

Вариант 12

Написать функцию, реализующую алгоритм сортировки вставками.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 13

Написать функцию, реализующую алгоритм Флойда – Уоршелла.

Входные значения: взвешенный граф, стартовый узел, целевой узел.

Выходные значения: кортеж, состоящий из дистанции между начальным и целевым узлом и путь из начального в целевой узел в виде словаря.

Вариант 14

Написать функцию, реализующую алгоритм Грэхема (построение выпуклой оболочки).

Входные значения: множество точек Q ($|Q| \geq 3$).

Выходные значения: список точек образующих выпуклую оболочку.

Вариант 15

Написать функцию, реализующую алгоритм A^* .

Входные значения: взвешенный граф, стартовый узел, целевой узел.

Выходные значения: кортеж, состоящий из дистанции между начальным и целевым узлом и путь из начального в целевой узел в виде словаря.

Вариант 16

Написать функцию, реализующую решето Сундарамы.

Входные значения: верхняя граница интервала $[1, N]$.

Выходные значения: список простых чисел в интервале $[1, 2N + 1]$.

Вариант 17

Написать функцию, реализующую метод Ньютона.

Входные значения: функция $f(x)$, производная функции $f(x) - f'(x)$, границы интервала поиска $[a, b]$.

Выходные значения: координаты минимума и значение функции в этой точке.

Вариант 18

Написать программу, реализующую ПИД-регулятор.

Входные значения: коэффициенты усиления пропорциональной K_p , интегрирующей K_i , дифференцирующей K_d составляющих регулятора; значение выхода объекта $x(t)$, значение уставки x^* .

Выходные значения: значение управления u .

По желанию студента реализовать фабрику регуляторов с заданными K_p, K_i, K_d .

Вариант 19

Написать функцию, реализующую алгоритм быстрой сортировки.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 20

Написать функцию, реализующую алгоритм Краскала.

Входные значения: взвешенный граф в виде хеш-таблицы (словаря).

Выходные значения: минимальное оставное дерево в виде словаря.

Вариант 21

Написать функцию, реализующую алгоритм сортировки подсчетом.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 22

Написать функцию, преобразующую строку с математическим выражением в обратную польскую запись.

Входные значения: строка с математическим выражением.

Выходные значения: строка с обратной польской записью.

Вариант 23

Написать функцию, реализующую Ро-алгоритм Полларда.

Входные значения: число N , которое необходимо разложить на множители; начальный элемент последовательности x_0 .

Выходные значения: список множителей числа N .

Вариант 24

Написать функцию, реализующую алгоритм Беллмана-Форда.

Входные значения: взвешенный граф, стартовый узел, целевой узел.

Выходные значения: кортеж, состоящий из дистанции между начальным и целевым узлом и путь из начального в целевой узел в виде словаря.

Вариант 25

Написать функцию, реализующую алгоритм сортировки прямым обменом.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 26

Написать функцию, реализующую алгоритм поиска Кнута-Морриса-Пратта.

Входные значения: строка, в которой необходимо произвести поиск; множество слов которые необходимо найти.

Выходные значения: словарь, ключи которого найденные слова, а значения – кортежи индексами начала и конца слова в исходной строке.

Вариант 27

Написать функцию, реализующую алгоритм сортировки слиянием.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 28

Написать функцию, реализующую алгоритм сортировки выбором.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 29

Написать функцию, реализующую алгоритм сортировки перемешиванием.

Входные значения: список длиной n .

Выходные значения: отсортированный список длиной n .

Вариант 30

Написать функцию, реализующую алгоритм кодирования на основе кода Хаффмана.

Входные значения: строка, которую необходимо сжать.

Выходные значения: исходная строка закодированная кодом Хаффмана.

Контрольные вопросы

1. Понятие области видимости. Четыре области видимости в Python.
2. Функции. Возвращаемые значения.
3. Замыкания и декораторы.

4. Ключевые аргументы, аргументы по умолчанию. Неименованные аргументы. Упаковка и распаковка аргументов.

5. Операторы `global` и `nonlocal`. Варианты применения.

6. Функциональное программирование в Python (`map`, `zip`, `filter`).
Анонимные функции.

7. Исключения. Метод обработки и возбуждения исключений.
Пользовательские исключения.

8. Нотация «*O*» большое. Цель определения сложности алгоритма.

9. Аннотация типов в Python, модуль `typing`.

10. Передача аргументов по ссылке и по значению.