

Теория для подготовки к экзамену.

1. Списки и операции над ними. Генераторы списков.
2. Строки и операции над ними.
3. Функции (описание, передача параметров, изменяемые и неизменяемые параметры, итеративные и рекурсивные функции, локальные и глобальные переменные, механизм возвращения значений функцией).
4. Файлы. Основные понятия. Файловые операции.
5. Матрицы. Основные понятия и операции над матрицами.
6. Словари и операции над ними. (it-направление)
7. Множества и операции над ними. (it-направление)
8. Основы ООП. Классы. Понятия атрибута и метода. Инкапсуляция, полиморфизм и наследование. Перекрытие операторов. (it-направление)

Задачи для подготовки к экзамену.

1. Получите список [3, 6, 9, 12, 15, ...] в одну строку. Количество элементов N вводится с клавиатуры.
2. Ввести список через пробел. Написать функцию, которая вернет список, в котором левая и правая части списка поменяны местами. Если в списке нечетное кол-во элементов, то центральный не трогать.
[1,2,3,4,5,6,7] -> [5,6,7,4,1,2,3]
3. С клавиатуры ввести список до нуля. Найти индекс максимального элемента.
4. Целочисленный список заполняется с клавиатуры через «точку». Проверьте, является ли он симметричным?
5. Заполнить список из n элементов случайными целыми числами из отрезка [a,b]. (a,b, n вводятся с клавиатуры) . Получить список в две строки.
6. С помощью генератора заполнить список словами, длина которых четна (слова в строке разделены пробелами). Строка вводится с клавиатуры.
7. Задан список L. Получите в одну строку список R, содержащий только четные и отрицательные элементы
8. Напишите функцию move, которая циклически сдвигает элементы списка L вправо:

$L[0] \rightarrow L[1], L[1] \rightarrow L[2] \dots L[N - 1] \rightarrow L[0]$

Использовать дополнительный список нельзя.

9. Список заполняется случайными целыми числами. Удалить все повторяющиеся элементы списка.
10. Написать функцию, которая получает матрицу и возвращает матрицу, у которой все элементы над побочной диагональю заменили на -1. Матрица записана в файле, результат записать в файл.
11. Написать программу, которая читает матрицу из файла и выводит количество строк, столбцов, последнюю строку и первый столбец.
12. Написать функцию, которая получает матрицу и возвращает матрицу, у которой все элементы под главной диагональю заменили на -1. Матрица записана в файле, результат записать в файл.
13. Написать функцию, которая получает матрицу и возвращает значение и индексы минимального элемента матрицы. Если минимальных элементов несколько, то вернуть первые индексы. Матрица записана в файле, результат записать в файл.
14. Написать функцию, которая получает две матрицы и возвращает разность элементов матрицы. Матрица записана в файле, результат записать в файл.
15. Написать функцию, которая получает две матрицы и возвращает их произведение. Матрицы записаны в файле, результат записать в файл.
16. Написать функцию, которая получает матрицу и возвращает транспонированную матрицу. Матрица записана в файле, результат записать в файл.
17. Написать функцию, которая получает матрицу, и возвращает матрицу, у которой последний и первый столбцы поменяли местами. Матрица записана в файле, результат записать в файл.
18. Написать функцию, которая получает матрицу, и возвращает матрицу, у которой первую и последнюю строки поменяли местами. Матрица записана в файле, результат записать в файл.
19. (it-направление) Напишите класс **Rational** – класс рациональных чисел таким, чтобы код, приведенный ниже, работал корректно. Примечание: дробь должна храниться без выделения целой части в сокращенном виде.

```
r1 = Rational(1, 2)
r2 = Rational(6, 4)
print(r1) # => 1/2
print(r2) # => 3/2
print(r1 + r2) # => 2/1
```

```
print(r1 - r2) # => -1/1
print(r1 + 2) # => 5/2
print(2 + r2) # => 7/2
print(r1 - 1) # => -1/2
print(2 - r2) # => 1/2
print(r1 * r2) # => 3/4
print(r1 / r2) # => 1/3
print(r1 * 2) # => 1/1
print(2 * r2) # => 3/1
print(r1 / 2) # => 1/4
print(1 / r2) # => 2/3
print(r1.whole()) # => 0
print(r2.whole()) # => 1
print(int(r1)) # => 0
print(int(r2)) # => 1
```