**Практикум. Функции**

## Простые задания на функции

**Задание 1. Паллиндром.** Необходимо определить, является ли это слово палиндромом (одинаково читается вперед и назад, например, АННА) с помощью функции. Если палиндром, на экран вывести True, иначе - False.

**Задание 2.** Объявите функцию с именем is\_triangle, которая принимает три стороны треугольника (целые числа) и проверяет, можно ли из переданных аргументов составить треугольник. (Напомню, что у любого треугольника длина третьей стороны всегда должна быть меньше суммы двух других). Если  проверка проходит, вернуть булево значение True, иначе - значение False.

Вызывать функцию не нужно, только задать.

**Входные данные**

3 4 5

**Выходные данные:**

True

**Задание 3.**Объявите функцию для проверки числа на нечетность (возвращается True, если переданное число нечетное и False, если число четное).

После объявления функции обработайте с её помощью список целых значений,И, используя генератор списков и созданную функцию, сформируйте список из нечетных значений на основе введенного исходного списка. Результат отобразите на экране командой:

print(\*lst)

где lst - сформированный список из нечетных значений.

**Входные данные**

st = [8, 11, -15, 3, 2, 10]

**Выходные данные:**

11 -15 3

**Задание 4.** Напишите функцию, которая проверяет корректность переданного ей email-адреса в виде строки. Будем полагать, что адрес верен, если он обязательно содержит символы '@' и '.', а все остальные символы могут принимать значения: 'a-z', 'A-Z', '0-9' и '\_'. Если email верен, то функция выводит ДА, иначе - НЕТ.

После объявления функции прочитайте (с помощью функции input) строку с email-адресом и вызовите функцию с этим аргументом.

**Входные данные**

sc\_lib@list.ru

**Выходные данные:**

ДА

Задание 5. Объявите функцию, которая принимает строку и заключает ее в указанный тег. Тег определяется формальным параметров tag с начальным значением в виде строки "h1". Например, мы передаем строку "Hello Python" и заключаем в тег "h1". На выходе должны получить строку (без кавычек):

"<h1>Hello Python</h1>"

## Задания на рекурсивные функции

**Задание 1.** Алгоритм вычисления значения функции F(n), где n  – натуральное число, задан следующими соотношениями:

F(1) = 1;

F(n) = F(n − 1) + n при n>1

Чему равно значение функции F(5), F(30)

Ответ: 465.

**Задание 2.** Последовательность чисел Фибоначчи задается рекуррентным соотношением:

F(1) = 1

F(2) = 1

F(n) = F(n–2) + F(n–1), при n >2, где n – натуральное число.

Чему равно восьмое число в последовательности Фибоначчи?

В ответе запишите только натуральное число.

Ответ: 21

**Задача 3.** (Сделать на основе первой задачи) Последовательность чисел трибоначчи задается рекуррентным соотношением:

F(1) = 0

F(2) = 1

F(3) = 1

F(n) = F(n–3) + F(n–2) + F(n–1), при n >3, где n – натуральное число.

Чему равно девятое число в последовательности трибоначчи?

В ответе запишите только натуральное число.

Ответ: 44

**Задача 4.** Алгоритм вычисления значения функции F(n), где n  – целое неотрицательное число, задан следующими соотношениями:

F(0) = 0;

F(n) = F(n / 2), если n > 0 и при этом чётно;

F(n) = 1 + F(n − 1), если n нечётно.

Сколько существует таких чисел n, что 1 ≤ n ≤ 1000 и F(n)  =  3?

Ответ: 120

**Задача 5.** У робота есть две команды, которым присвоены номера:

1.  Прибавить 1

2.  Прибавить 3

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 3. Программа для робота – это последовательность команд.

Сколько существует программ, для которых при исходном числе 1 результатом является число 17 и при этом траектория вычислений содержит число 9? Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы 121 при исходном числе 7 траектория будет состоять из чисел 8, 11, 12.

Ответ: 169

## Задания на lamda, filter

**Задача 1.** На складе лежат мешки различного веса и стоимости. Вес за мешок и стоимость записаны на как натуральные числа с списке. Отсортировать мешки по стоимости за кг, при равной стоимости первыми идут мешки с наименьшим весом

S = [  
 [47, 470],  
 [50, 600],  
 [60, 480],  
 [45, 540],  
 [30, 300]  
]

**Задача 2.** Отфильтровать стоп-слова (союзы и предлоги) из строки.

# Строка со стоп-словами

stop\_list = ["в", "и", "по", "за", "на", ".", ",", "г", ":"]

string='''

Сервис по поиску работы и сотрудников

HeadHunter опубликовал подборку

высокооплачиваемых вакансий в России за ноябрь 2024 года

в Москве : на первых строчках IT-архитекторы и техлиды

'''

**Задача 3.** Написать программу, чтобы посчитать площадь квартиры, состоящей из комнат rooms. Попробуйте применить map и reduce

rooms = [

    {"name": "кухня", "length": 6, "width": 4},

    {"name": "комната 1", "length": 5.5, "width": 4.5},

    {"name": "комната 2", "length": 5, "width": 4},

    {"name": "комната 3", "length": 7, "width": 6.3},

]

**Задача 4.** Лишний символ. Робот отправляет сообщения. Есть только две проблемы – иногда символы в каждом сообщении перемешиваются, и всегда добавляется ровно один лишний символ. Напишите функцию, которая принимает две строки – исходное корректное сообщение и поврежденное доставленное, – и находит лишний символ.

Формат ввода:

Строка А – исходное сообщение.

Строка Б – доставленное сообщение.

#Пример ввода:

スーパーハカー

スーパーハッカー

Формат вывода:

Лишний символ в доставленном сообщении.

#Пример вывода:

ッ