Струков Артемий Викторович

МИИГАиК 2023-ФГиИБ-ПИ-1б

**Отчет по выполненным заданиям по**

**предмету “Алгоритмизация и программирование”**

**Задача 1**

**Постановка задачи:** Программа принимает на вход натуральное число **n**. Напечатать все квадраты чисел, не превосходящие число **n**.

**Пример**:

Входные данные: 50

Выходные данные: 1 4 9 16 25 36 49

**Решение:**

n = int(input())  
for num in range(1, int(n\*\*0.5)+1):  
 print(num\*\*2, end=" ")

**Задача 2**

**Постановка задачи:** Программа принимает на вход число **n**. Напечатать сумму чисел **n**, пока **n** не равно 0.

**Пример:**

Входные данные: 1 2 3 4 5 0

Выходные данные: 15

**Решение:**

n = int(input())  
while n != 0:  
 new\_n = int(input())  
 if new\_n == 0:  
 break  
 n += new\_n  
print(n)

**Задача 3**

**Постановка задачи:** Последовательность состоит из натуральных чисел. Определить значение второго по величине в этой

**Пример:**

Входные данные: 1 2 3 9 0 12 3

Выходные данные: 9

**Решение:**

data = list(map(int, input().strip().split(" ")))  
data.sort(reverse=True)  
print(data[1])

**Задача 4**

**Постановка задачи:** Последовательность состоит из натуральных чисел (**n**-ок). Определить какое наибольшее число подряд, идущий элементов последовательности равны друг другу. Последовательность определяется до момента, пока **n** не равно 0.

**Примеры:**

Входные данные: 1 1 1 2 2 1 1

Выходные данные: 3

Входные данные: 2 1 1 2 2 1 1

Выходные данные: 2

Входные данные: 2 1 1 2 2 2 1 1

Выходные данные: 3

Входные данные: 1 2 3 4 5 6 7

Выходные данные: 1

**Решение:**

data = list(map(int, input().strip().split(" ")))  
max\_, current = 1, 1  
for i in range(1, len(data)):  
 if data[i] == data[i-1]:  
 current += 1  
 else:  
 max\_ = max(max\_, current)  
 current = 1  
print(max\_)

**Задача 5**

**Постановка задачи:** Определить наименьшее расстояние между двумя локальными максимумами последовательности натуральных чисел, завершающихся числом 0. Если в последовательности нет 2-х локальных максимумов, то выведите число 0. Начальное и конечные значения не являются локальными максимумами. Расстоянием считается кол-во пробелов между элементами.

**Примеры:**

Входные данные: 1 2 3 4 5 0

Выходные данные: 0

Входные данные: 1 2 1 1 2 1 0

Выходные данные: 3

Входные данные: 1 2 3 2 1 0

Выходные данные: 0

Входные данные: 1 2 3 2 1 3 0

Выходные данные: 3

Входные данные: 1 2 3 2 1 3 2 3 0

Выходные данные: 2

**Решение:**

data = list(map(int, input().strip().split(" ")))  
indexes\_local\_maximum = []  
for index in range(1, len(data)-1):  
 current\_value = data[index]  
 prev\_value = data[index-1]  
 next\_value = data[index+1]  
 if prev\_value < current\_value and current\_value > next\_value:  
 indexes\_local\_maximum.append(index)  
  
if len(indexes\_local\_maximum) < 2:  
 print(0)  
else:  
 min\_ = len(data)  
 for i in range(1, len(indexes\_local\_maximum)):  
 current = indexes\_local\_maximum[i]  
 prev = indexes\_local\_maximum[i-1]  
 diff = current - prev  
 min\_ = min(min\_, diff)  
  
print(min\_)

**Задача 6**

**Постановка задачи:** Дан список, переставьте соседние элементы в нём. Если элементов нечетное кол-во, то последний элемент остается на своём месте.

**Примеры:**

Входные данные: 1 2 3 4 5 6

Выходные данные: 2 1 4 3 6 5

Входные данные: 1 2 3 4 5

Выходные данные: 2 1 4 3 5

Входные данные: 1 3 4 2 9 0 1 2 7 8

Выходные данные: 3 1 2 4 0 9 2 1 8 7

**Решение:**

data = input().split(" ")  
for i in range(len(data)):  
 if (i+1) % 2 != 0:  
 print(data[i+1] if len(data)-1 > i else data[i], end=" ")  
 else:  
 print(data[i-1], end=" ")

**Задача 7**

**Постановка задачи:** Дан список, упорядоченный по возрастанию элементов. Определите сколько в нем различных элементов. В случае ввода числа, не по условию – выводиться ошибка и это число не учитывается

**Примеры:**

Входные данные: 1 2 3 4 5 6 7

Выходные данные: 7

Входные данные: 1 2 2 3 4

Выходные данные: 4

Входные данные: 1 2 3 2 2 4

Выходные данные: 3

**Решение:**

last = None  
count = 0  
while True:  
 inp = input()  
 if inp == "":  
 break  
 num = int(inp.strip())  
 if last is not None and last > num:  
 print(f"{num} не подходит, необходимо число >= {last}")  
 else:  
 if last != num:  
 count += 1  
 last = num  
  
print(count)

**Задача 8**

**Постановка задачи:** Вводится размер массива, затем и сам массив. Необходимо вывести элементы массива встречающийся только один раз и в порядке их ввода.

**Примеры:**

Входные данные:

3

1 2 3

Выходные данные: 1 2 3

Входные данные:

6

1 2 3 3 0 1

Выходные данные: 2 0

**Решение:**

n = int(input())  
data = input().split(" ")  
for i in data:  
 if data.count(i) == 1:  
 print(i, end=" ")