

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №1
дисциплины «Алгоритмизация»

Выполнила:
Середа Кирилл Витальевич
1 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника», очная
форма обучения

(подпись)

Руководитель практики:
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Ход выполнения заданий

1) Задание 1(58228):

Алгоритм вычисления значения функции $F(n)$, где n — натуральное число, задан следующими соотношениями:

$F(n) = 1$ при $n = 1$;

$F(n) = 2$ при $n = 2$;

$F(n) = \left\lfloor \frac{4 \cdot n - F(n-3)}{8} \right\rfloor$, если $n > 2$ и при этом если n чётно;

$F(n) = \left\lfloor \frac{4 \cdot n - F(n-1) + F(n-2)}{8} \right\rfloor$, если $n > 2$ и при этом n нечётно.

Чему равно значение функции $F(52) - F(38)$?

Примечание: квадратные скобки в записи $\lfloor x \rfloor$ применяются для обозначения целой части числа x .

Рисунок 1 – Условие задания

```
def F(n):
    if n == 1: return 1
    elif n == 2: return 2
    elif n < 1: return 0
    elif (n > 2) and (n%2 != 0): return (4*n - F(n-3))/8
    elif (n > 2) and (n%2 == 0): return (4*n - F(n-1) + F(n-2))/8
ANS = F(52)-F(38)
print(ANS)
F() > elif n < 1
16
/Users/MelancholySeal/PycharmProjects/heyhumanity/venv/bin/python /Users/MelancholySeal/PycharmProjects/heyhumanity/16.py
7.127272727272473
Process finished with exit code 0
```

Рисунок 2 – Код программы и результат выполнения

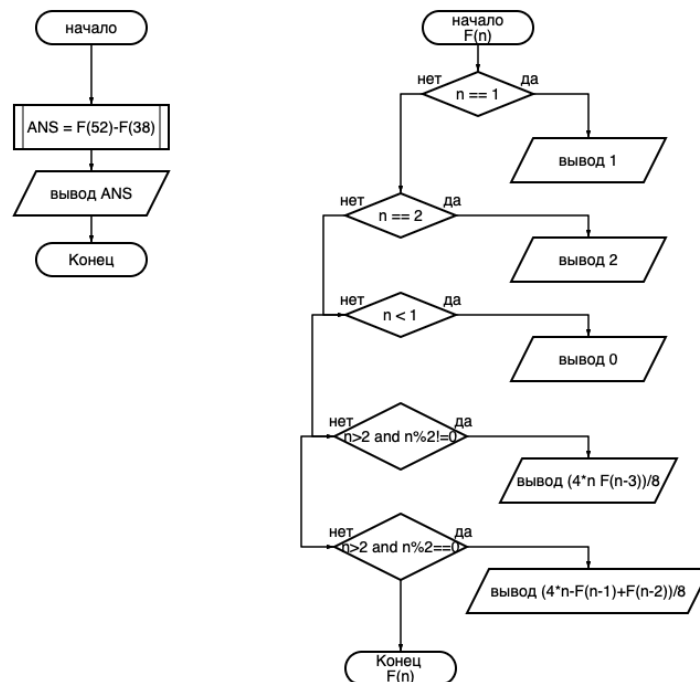


Рисунок 3 – Блок-схема для задания 1

2) Задание 2(37348): В файле содержится последовательность из 10 000 целых положительных чисел. Каждое число не превышает 10 000. Определите и запишите в ответе сначала количество пар элементов

последовательности, для которых произведение элементов не кратно 34, затем максимальную из сумм элементов таких пар. В данной задаче под парой подразумевается два различных элемента последовательности. Порядок элементов в паре не важен.

```
count,m = 0,0
f = open('17.txt')
arr = [int(i) for i in f]
for i in range(len(arr) - 1):
    for j in range(i + 1, len(arr)):
        if arr[i] * arr[j] % 34 != 0:
            count += 1
            m = max(m, arr[i] + arr[j])
print(count, m)
```

17 x

/Users/MelancholySeaL/PycharmProjects/heyhumanit
45539445 19999

Рисунок 4 – Код программы и результат ее выполнения

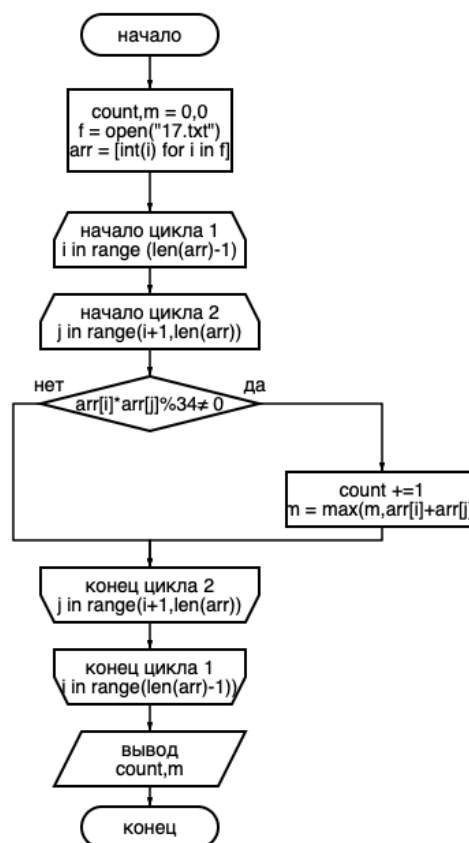


Рисунок 5 – Блок-схема для задания 2

3) Задание 3(27415): Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз — в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата лежит монета достоинством от 1 до 100. Посетив клетку, Робот забирает монету с собой; это также относится к начальной и конечной клетке маршрута Робота.

Откройте файл. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответ запишите два числа друг за другом без разделительных знаков — сначала максимальную сумму, затем минимальную. Исходные данные представляют собой электронную таблицу размером $N \times N$, каждая ячейка которой соответствует клетке квадрата.

```
import openpyxl

workbook = openpyxl.load_workbook('18.xlsx')

sheet = workbook.active

N = sheet.max_row

coins = [[0] * N for _ in range(N)]

for i in range(N):
    for j in range(N):
        coins[i][j] = sheet.cell(row=i + 1, column=j + 1).value

max_coins = [[0] * N for _ in range(N)]
min_coins = [[0] * N for _ in range(N)]

max_coins[0][0] = min_coins[0][0] = coins[0][0]

for i in range(1, N):
    max_coins[i][0] = min_coins[i][0] = max_coins[0][i] = min_coins[0][i] = max_coins[0][i - 1] + coins[0][i]

for i in range(1, N):
    for j in range(1, N):
        max_coins[i][j] = max(max_coins[i - 1][j], max_coins[i][j - 1]) + coins[i][j]
        min_coins[i][j] = min(min_coins[i - 1][j], min_coins[i][j - 1]) + coins[i][j]

max_sum = max_coins[N - 1][N - 1]
min_sum = min_coins[N - 1][N - 1]

workbook.close()

print(max_sum, min_sum)
for i in range(1, N):
    for j in range(1, N):
```

18

1204 502

Рисунок 6 – Код программы и результат ее выполнения

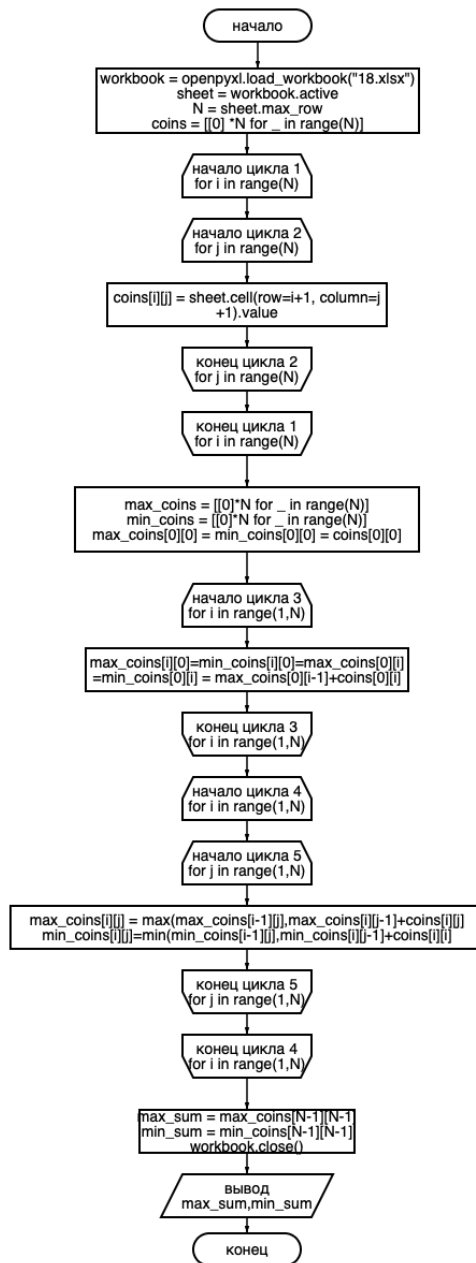


Рисунок 7 – Блок-схема для задания 3