

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»**  
**Вариант 29**

Выполнил:  
Саенко Андрей Максимович  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение средств  
вычислительной техники и  
автоматизированных систем», заочная  
форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Р.А., канд. технических  
наук, доцент кафедры  
инфокоммуникаций

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_  
Ставрополь, 2023 г.

Порядок выполнения работы:

### Задание 1

Алгоритм вычисления значения функции  $F(n)$ , где  $n$  - целое неотрицательное число, задан следующими соотношениями:

$F(n) = 0$ , при  $n \leq 1$ ;

$F(n) = F(n - 1) + 3n^2$ , если  $n > 1$  и при этом нечётно;

$F(n) = n / 2 + F(n - 1) + 2$ , если  $n > 1$  и при этом чётно.

Чему равно значение функции  $F(49)$ ? В ответе запишите только целое число.

Блок-схема программы:

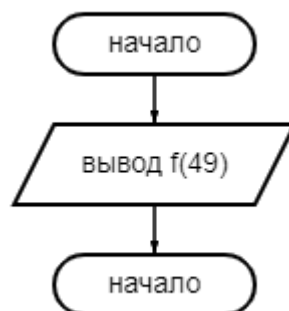
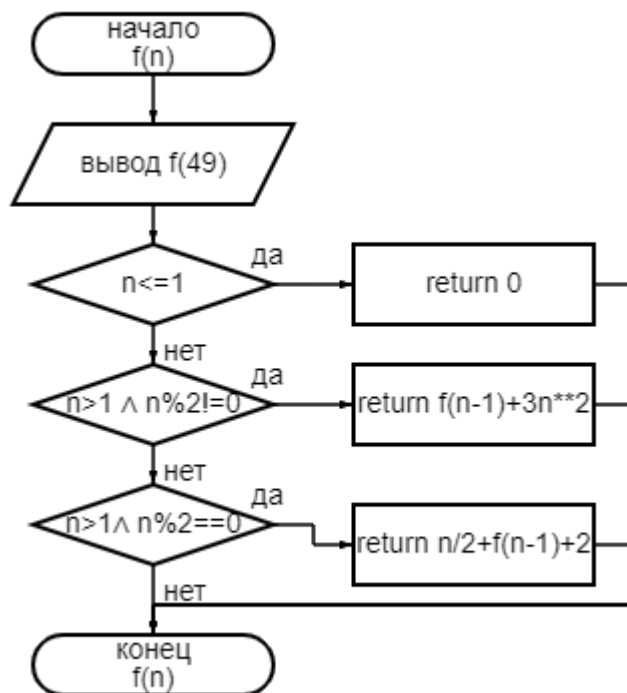
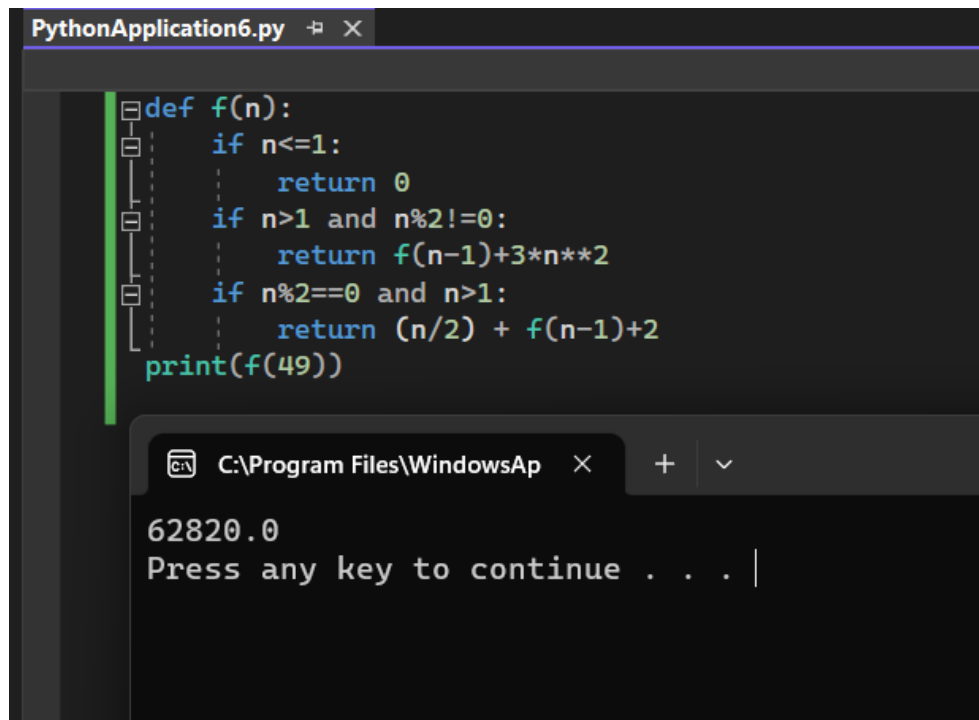


Рисунок 1. Блок-схема программы для задания 1

Код программы:

```
def f(n):  
    if n<=1:  
        return 0  
    if n>1 and n%2!=0:  
        return f(n-1)+3*n**2  
    if n%2==0 and n>1:  
        return (n/2) + f(n-1)+2  
print(f(49))
```

Результат работы программы:

The image shows a screenshot of a Python application window titled "PythonApplication6.py". The window has a dark background with light-colored text. The code is displayed in a monospaced font. The code defines a function f(n) and calls it with the argument 49. The function f(n) is a recursive function that calculates a value based on the parity of n. The output of the program is displayed in a separate window at the bottom, showing the value 62820.0 and a prompt to press any key to continue. The code in the window is:

```
def f(n):  
    if n<=1:  
        return 0  
    if n>1 and n%2!=0:  
        return f(n-1)+3*n**2  
    if n%2==0 and n>1:  
        return (n/2) + f(n-1)+2  
print(f(49))
```

The output window shows:

```
62820.0  
Press any key to continue . . . |
```

Рисунок 2. Результат работы программы

## Задание 2

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, у которых разность элементов кратна 60 и хотя бы один из элементов кратен 15, затем максимальную из разностей элементов таких пар. В данной задаче под парой подразумевается два различных элемента последовательности. Порядок элементов в паре не важен.

Блок-схема программы:

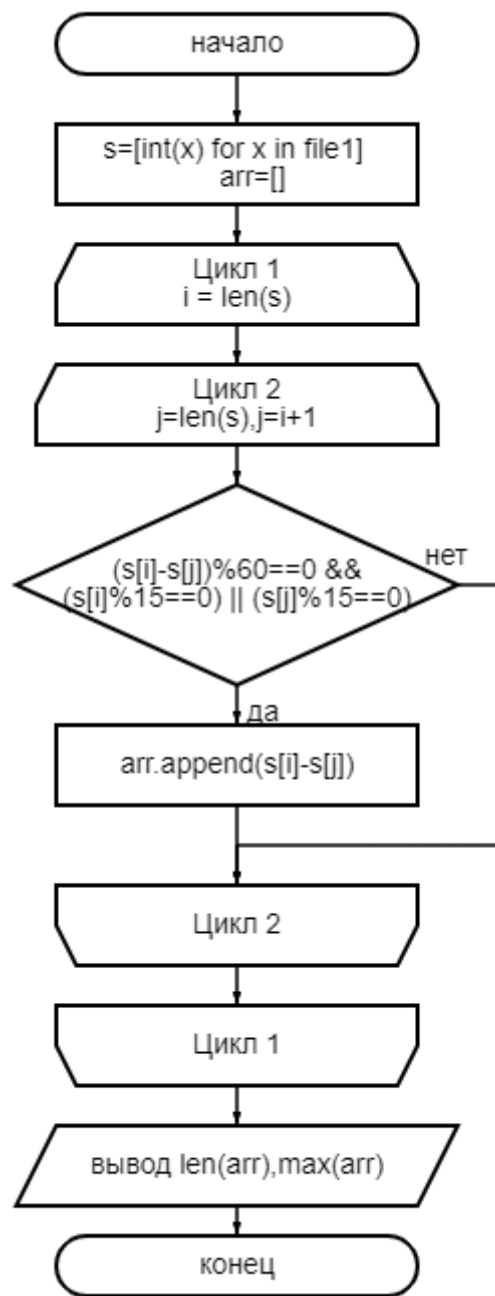
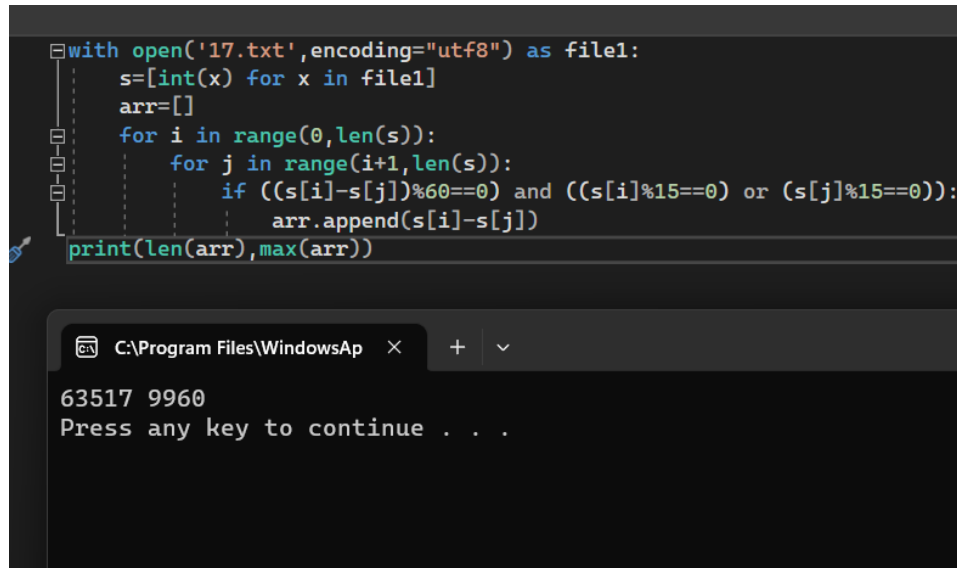


Рисунок 3. Блок-схема программы для задания 2

Код программы:

```
with open('17.txt', encoding="utf8") as file1:
    s=[int(x) for x in file1]
    arr=[]
    for i in range(0, len(s)):
        for j in range(i+1, len(s)):
            if ((s[i]-s[j])%60==0) and ((s[i]%15==0) or (s[j]%15==0)):
                arr.append(s[i]-s[j])
print(len(arr), max(arr))
```

Результат работы программы:

The image shows a screenshot of a code editor and a terminal window. The code editor displays a Python script that reads a file named '17.txt', processes its contents into a list 's', and then iterates through pairs of elements (s[i], s[j]) where i < j. It checks if the difference (s[i] - s[j]) is divisible by 60 and if either s[i] or s[j] is divisible by 15. If both conditions are met, the difference is added to a list 'arr'. Finally, it prints the length of 'arr' and its maximum value. The terminal window shows the output '63517 9960' and a prompt 'Press any key to continue . . .'.

```
with open('17.txt', encoding="utf8") as file1:
    s=[int(x) for x in file1]
    arr=[]
    for i in range(0, len(s)):
        for j in range(i+1, len(s)):
            if ((s[i]-s[j])%60==0) and ((s[i]%15==0) or (s[j]%15==0)):
                arr.append(s[i]-s[j])
print(len(arr), max(arr))
```

C:\Program Files\WindowsAp x + v

63517 9960  
Press any key to continue . . .

Рисунок 4. Результат работы программы

### Задание 3

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 вверх. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вверх - в соседнюю верхнюю. 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 нижней клетки в правую верхнюю. 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.

## Блок-схема программы:

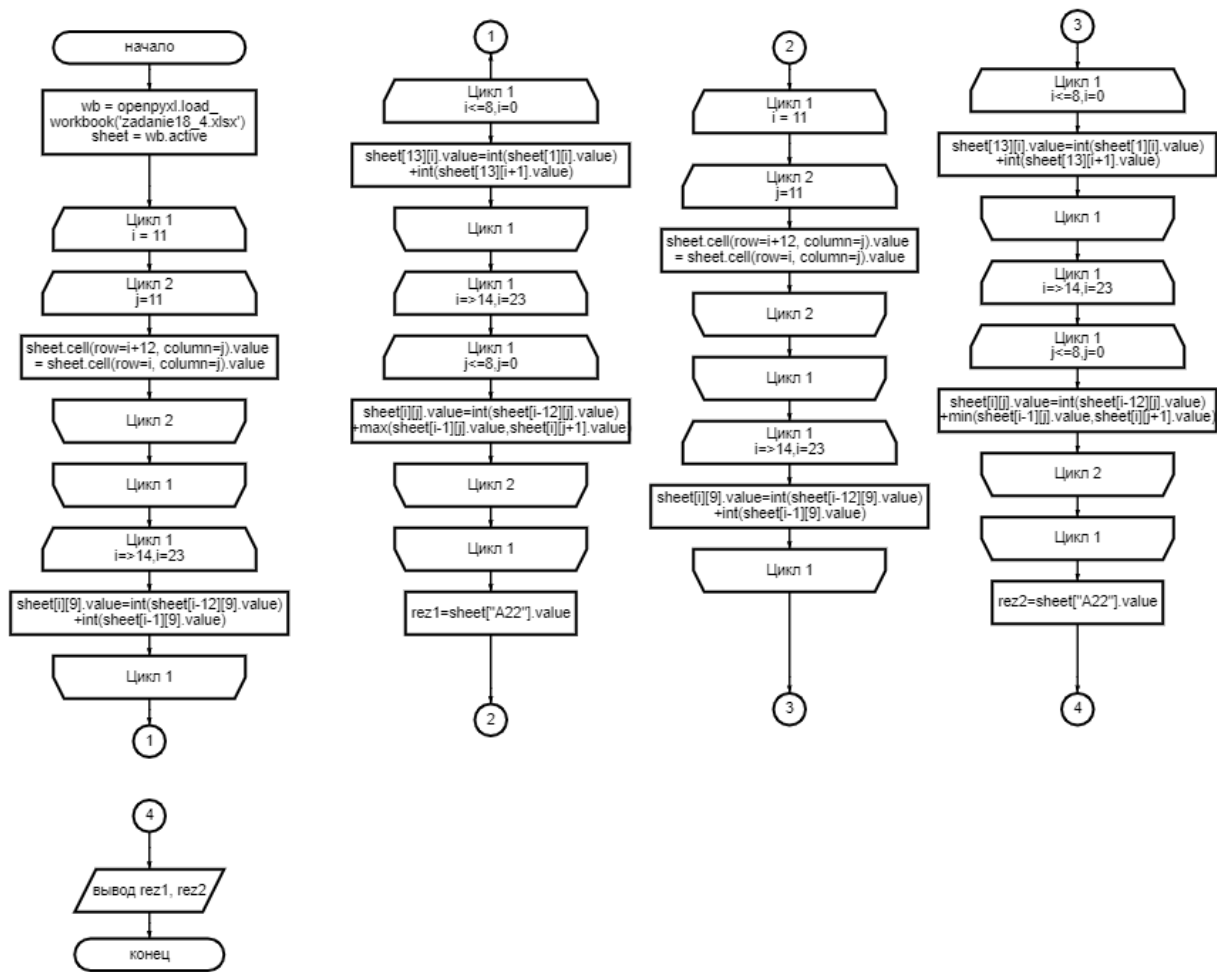


Рисунок 5. Блок-схема программы для задания 3

## Код программы:

```

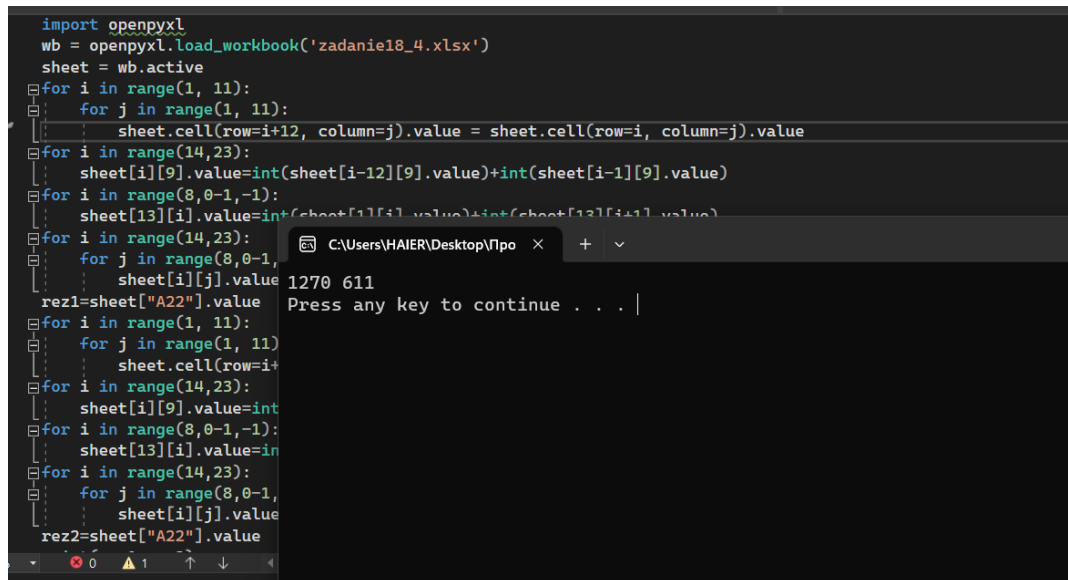
import openpyxl
wb = openpyxl.load_workbook('zadanie18_4.xlsx')
sheet = wb.active
for i in range(1, 11):
    for j in range(1, 11):
        sheet.cell(row=i+12, column=j).value = sheet.cell(row=i, column=j).value
for i in range(14, 23):
    sheet[i][9].value = int(sheet[i-12][9].value) + int(sheet[i-1][9].value)
for i in range(8, 0-1, -1):
    sheet[13][i].value = int(sheet[1][i].value) + int(sheet[13][i+1].value)
for i in range(14, 23):
    for j in range(8, 0-1, -1):
        sheet[i][j].value = int(sheet[i-12][j].value) + max(sheet[i-1][j].value, sheet[i][j+1].value)
rez1 = sheet["A22"].value
for i in range(1, 11):
    for j in range(1, 11):
        sheet.cell(row=i+12, column=j).value = sheet.cell(row=i, column=j).value
for i in range(14, 23):
    sheet[i][9].value = int(sheet[i-12][9].value) + int(sheet[i-1][9].value)
for i in range(8, 0-1, -1):
    sheet[13][i].value = int(sheet[1][i].value) + int(sheet[13][i+1].value)
  
```

```

for i in range(14,23):
    for j in range(8,0-1,-1):
        sheet[i][j].value=int(sheet[i-12][j].value)+min(sheet[i-1][j].value,sheet[i][j+1].value)
rez2=sheet["A22"].value
print(rez1,rez2)
wb.save('zadanie18_4.xlsx')

```

Результат работы программы:



```

import openpyxl
wb = openpyxl.load_workbook('zadanie18_4.xlsx')
sheet = wb.active
for i in range(1, 11):
    for j in range(1, 11):
        sheet.cell(row=i+12, column=j).value = sheet.cell(row=i, column=j).value
for i in range(14,23):
    sheet[i][9].value=int(sheet[i-12][9].value)+int(sheet[i-1][9].value)
for i in range(8,0-1,-1):
    sheet[13][i].value=int(sheet[11][i].value)+int(sheet[12][i+1].value)
for i in range(14,23):
    for j in range(8,0-1,-1):
        sheet[i][j].value
rez1=sheet["A22"].value
for i in range(1, 11):
    for j in range(1, 11):
        sheet.cell(row=i+12, column=j).value = sheet.cell(row=i, column=j).value
for i in range(14,23):
    sheet[i][9].value=int(sheet[i-12][9].value)+int(sheet[i-1][9].value)
for i in range(8,0-1,-1):
    sheet[13][i].value=int(sheet[11][i].value)+int(sheet[12][i+1].value)
for i in range(14,23):
    for j in range(8,0-1,-1):
        sheet[i][j].value
rez2=sheet["A22"].value
1270 611
Press any key to continue . . . |

```

Рисунок 6. Результат работы программы

## Вывод

Были написаны программы на языке Python для выполнения заданий по программированию.