Міністерство освіти та науки України

Національний технічний університет України

«Київський Політехнічний Інститут ім. Ігоря Сікорського»

ФТІ

Кафедра ФТЗЗІ

**Лабораторна робота №6**

з дисципліни: «Програмування 4»

на тему:

**«Обмін інформацією з файлами та модульний принцип програмування»**

**Варіант 21**

Виконав:

Ст. гр. ФЕ-81

Пелих Валентин

Перевірив:

доцент Прогонов Д.О.

Київ 2020

Мета роботи: Оволодіти навичками роботи з файлами даних та засвоїти модульний принцип програмування.

**1. Порядок виконання роботи**

1. Проаналізувати умову задачі.

2. Розробити алгоритм та створити програму розв’язання задачі згідно з номером варіанту. При цьому розмістити функції програми у двох (або більше) різних файлах, та об’єднати ці файли у спільну програму двома різними способами:   
1) реалізувати модульний принцип програмування, створивши проект, у якому всі функції, крім головної, містяться в окремому файлі;

3. Результати роботи оформити протоколом.

**2. Варіанти завдань**

Модифікувати створену під час виконання у лабораторній роботі № 5 програму згідно з номером варіанту, використовуючи потоки файлового вводу/виводу для зчитування вхідної інформації з одного файла та запису отриманого результату в інший.

Код реалізації:

Functions.py:

class Exception(Exception):

def init(self, text):

self.txt = text

def check\_perfect\_square(a):

if (a == 1):

print("Processing Error!")

raise Exception("The 'a' must not equal to 1!")

x = a // 2

seen = set([x])

while x \* x != a:

x = (x + (a // x)) // 2

if x in seen:

return False

seen.add(x)

return True

def f(n, a, c):

def g(n):

res = (a\*n + a\*c) % 10

return res

if (0 <= n <= 9):

return n

else:

res = g(n) \* f((n-1-g(n)), a, c) + n

return res

Main.py

import functions as f

#test

#d1 = open("/Users/iced\_flavour/Desktop/solution/data1.txt", "r")

#d2 = open("/Users/iced\_flavour/Desktop/solution/data2.txt", "r")

#git

d1 = open("data1.txt", "r")

d2 = open("data2.txt", "r")

print("Lab 5.1", "\n")

try:

N = int(d1.readline())

an = []

print("Enter non-negative numbers to sequence.")

for i in range(0, N):

i = int(d1.readline())

an.append(i)

print("Your sequence is: ")

print(an, "\n")

print("Checking if some elements here is perfect square...")

for i in range(0, N):

print(an[i], "-", f.check\_perfect\_square(an[i]), end="; ")

print("\n")

segments = []; max\_seg = []

flow = False; a = 0

for i in range(0, N):

check = f.check\_perfect\_square(an[i])

if (check == True):

segments.append(an[i])

flow = True

else:

segments.append(0)

flow = False

if (flow == True):

a = a + 1

elif (flow == False):

max\_seg.append(a)

a = 0

max\_seg.append(a)

print("Max Segments are:", max\_seg, "\n")

max\_s = max(max\_seg)

print("The largest segment with perfect squares is: ", max\_s)

print("\n")

print("Lab 5.2", "\n")

print("Enter a, c, m. They must be natural numbers (positive integers).")

print("Otherwise - it will be converted using abs().")

a = abs(int(d2.readline()))

c = abs(int(d2.readline()))

m = abs(int(d2.readline()))

Res = f.f(m, a, c)

print("Result is: ", Res)

except f.Exception as non1:

print(non1)

except ValueError:

print("Entered elements must be non-negative!", "\n")

Приклади виконання програми:

