**Лабораторна робота №10 Вступ до документації проекту**

1 Вимоги

1.1**Розробник**

-Носов Микола

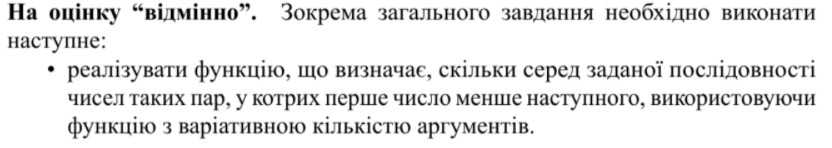
-студент групи кіт 120б

-8.12.2020

**1.2 Мета:**

Навчитися писати повноцінні звіти у двох форматах doc та markdown (звіт стосується 7 лабораторної роботи)

1.3 **Загальне завдання**



2 Опис програми

**2.1 Функціональне призначення**

Програма призначена для**:**

-знаходження НОД 2 чисел;

-перевірки на білета на “щастливість”;

-перевірки числа, і визначення, чи є воно простим;

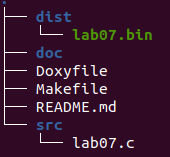
-перевірки числа, і визначення, чи є ідеальним;

-знаххождення квадратного кореня числа;

-центрування слів у строчці;

**2.2 Логічна структура проекту**

Структура проекту (див рис. 1)



*Рисунок 1* - *робота утиліти tree*

1. Функція знаходження НОД (див рис. 2)



*Рисунок 2 - фрагмент коду*

**Призначення :**

-Знаходження найбільшого спільного дільника ;

**Аргументи :**

-2 числа;

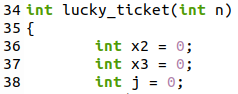
**Опис роботи :**

**-**визначення, яке число число є більшим;

**-**ділення більшого числа на меньше по %, доки вони ≠ 0;

**-**сумуємо 2 чилса;

1. Функція визначення щасливого білетика (див рис. 3)



*Рисунок 3 - фрагмент коду*

**Призначення :**

-Визначення, чи є білетик щасливим;

**Аргументи :**

-6-ти значне число;

**Опис роботи :**

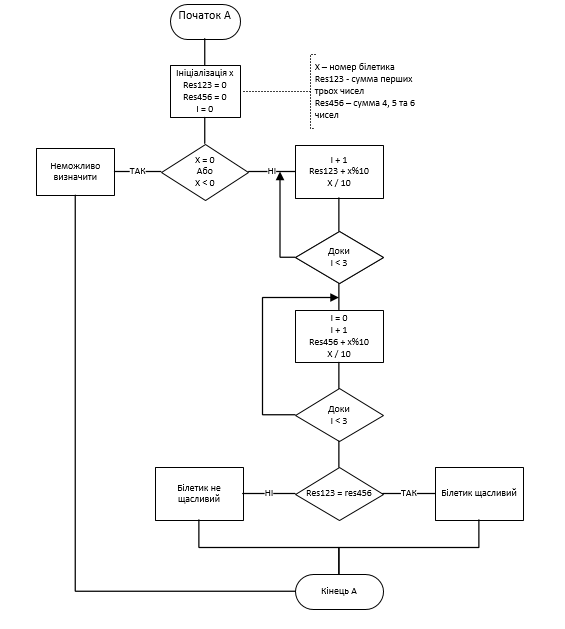
-розбиття 6-ти значного числа на 6 цифр;

-сума перших трьох чисел;

-сума чисел, що залишилися;

-якщо суми однокові - білетик щасливий;

**Схема алгоритму** (див рис 4) :



*Рисунок 4 - креслення алгоритму*

1. Функція визначення простого числа (див рис. 5)



*Рисунок 5 - фрагмент коду*

**Призначення :**

-Визначення, чи є число простим;

**Аргументи :**

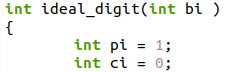
-Число

**Опис роботи :**

*-*Ділити число на всі числа від 1 до квадратного кореня заданого числа;

-Якщо число ділиться на ціло - число не просте, в іншому випадку - просте;

1. Функція визначення ідеально числа (див рис.6)



*Рисунок 6 - фрагмент коду*

**Призначення :**

-Визначення, чи є число ідеальним;

**Аргументи :**

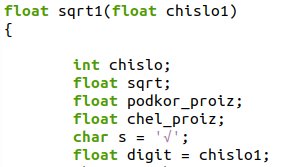
-Число;

**Опис роботи :**

*-*знаходження дільників числа;

-якщо сума дільників = самому числу - воно ідеальне;

1. Функція знахождення квадратного кореня (див рис 7)



*Рисунок 7 - фрагмент коду*

**Призначення :**

-Знахождення квадратного кореня;

**Аргументи :**

-Число

**Опис роботи :**

*-*знаходження дільників числа;

*-*попарне винесення дільників числа із під знаку кореня;

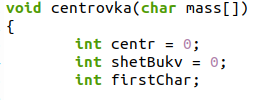
-знаходження кореня за теоремою дотичних Ньютона;

Отримаємо відповідь у такому вигляді (див рис 8 ):



*Рисунок 8 - квадратний корінь з числа 200*

1. Функція центрування слова у рядку (див рис 9)



*Рисунок 9 - функція, що нічого не повертає*

**Призначення :**

-Центрування слова або словосполучення у рядку;

**Аргументи :**

-Слово, що потребує цетрування;

**Опис роботи :**

*-*Знаходження першої букви;

-Підрахунок кількості букв;

-Розрахунок необхідної кількости символів центрування;

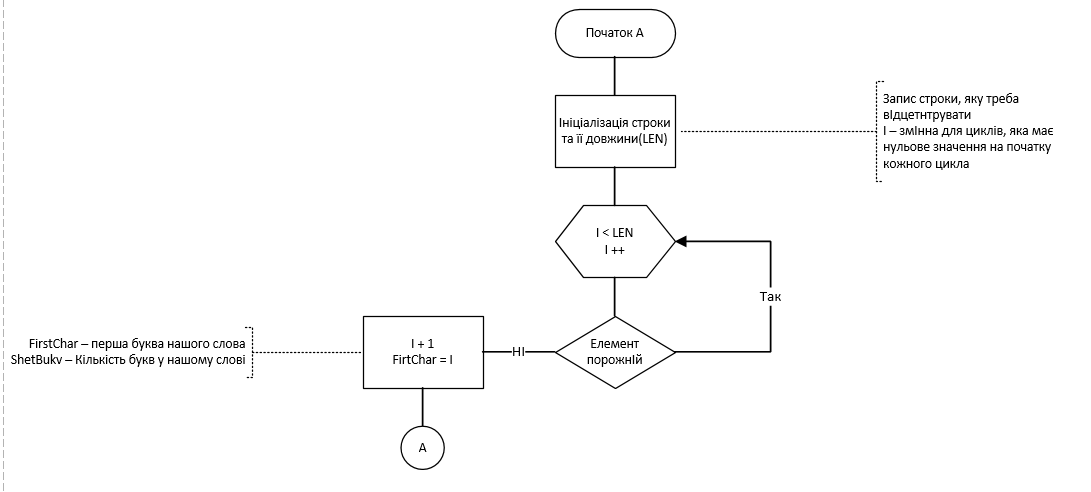
-Заповнення рядка буквами та символами “\_”;

Отримаємо відповідь у такій формі (див рис 10)

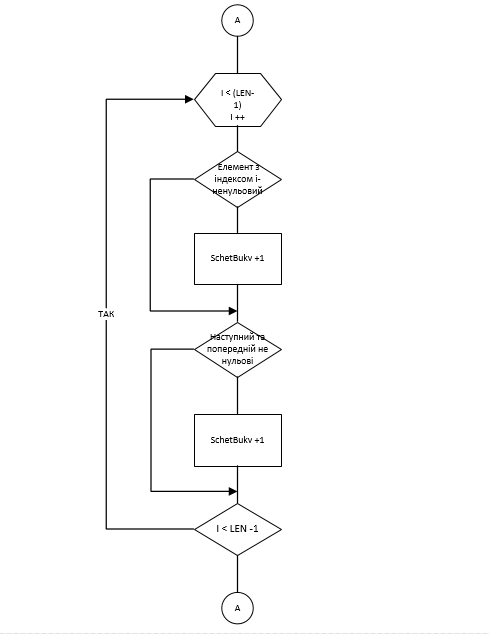


*Рисунок 10 - результат роботи функції 6*

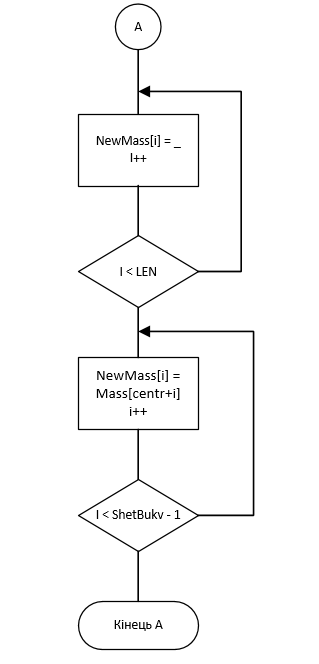
**Схем алгоритму** (див рис 11, 12, 13):



*Рисунок 11 - 1 частина креслення алгоритму*

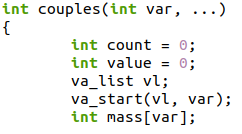


*Рисунок 12 - 2 креслення алгоритму*

**

*Рисунок 13 - 3 частина креслення алгоритму*

7 Функція з варіативною кількістю агрументів (див рис 14)



*Рисунок 14 - фрагмент коду*

**Призначення :**

-Знаходження кількості пар чисел, перше число якої меньше за наступне;

**Аргументи :**

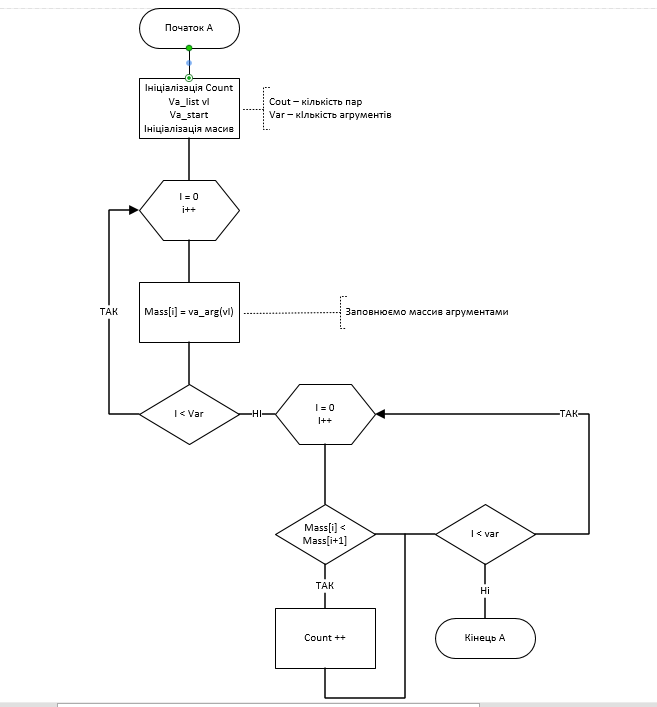
**-**Ряд чисел;

**Опис роботи :**

*-*Программа порівнює 2 числа, що стоять поруч;

*-*Якщо число меньше наступного - образовується пара;

**Схема алгоритму** (див рис 15):



*Рисунок 15 - креслення алгоритму*

**3 Варіанти використання**

Для обчислення результатів мы використовємо gdb - дебагер, влаштований в

утиліту gcc.

Щоб подивитися на резутат, ми повинні:

-при компіляції вказати рівень інформації для відлатки за допомогою -g; -відкомпілювати наш файл з кодом;

-запустити його бінарний файл у відлагоднику;

-вказати им’я функції, як точку зупину щоб подивитьсь на значення змінних;

-запускаємо програму та вводимо: “info locals”;

**Висновок:**

Протягом виконання даної лабораторної роботи ми навчилися розбивати нашу програму на підпрограми за допомогою функцій, а також познайомились з функцією, яка має варіативну кількість агрументів.

А також навчилися писати документацію згідно с ГОСТами та у Markdown.