Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний

інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 5 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження складних циклічних алгоритмів»

Варіант\_**20**\_\_

Виконав студент: **ІП-11 Лошак Віктор Іванович**

(шифр, прізвище, ім'я, по батькові)

Перевірив: **Мартинова О.П.**

Київ 2021

**Лабораторна робота 5**

**Дослідження складних циклічних алгоритмів**

**Мета:**

– дослідити особливості роботи складних циклів та набути практичних навичок їх

використання під час складання програмних специфікацій

**Варіант №20**

**Задача:**

Знайти цифровий корінь числа n

**Розв’язок:**

1)Постановка задачі

Результатом розв’язку є число значення якого отримується

із цифр заданого користувачем числа шляхом ітераційного процесу знаходження спочатку суми цифр даного числа, а потім, якщо потрібно, суми цифр значень, отриманих на попередній ітерації знаходження відповідних сум (якщо значення суми не є цифрою). Для обчислення виразу при заданому n використаємо арифметичний цикл. Значенням n є число що вводиться користувачем за запитом на початку програми. Для правильного виконання програми робимо перевірку числа на знак, і у випадку якщо число відємне конвертуємо його в додатнє. Зовнішній цикл буде виконуватись до того часу поки отриманий у внутрішньому циклі результат не стане однорозрядним числом. Загалом зовнішній цикл містить два внутрішнійх, перший з яких обраховує кількість цифр в заданому числі а другий обчислює їх суму. Для реалізації алгоритму використаємо арифметичні та логічні цикли також логічні оператори. Для використання математичних функцій степеня використаємо бібліотеки <cmath> для С++ і math для python.

2)Побудова математичної моделі

Складемо таблицю імен змінних

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Користувацький ввід | цілий | n | Початкове дане/ результат обчислень |
| Неконстантна копія значення n | цілий | copy | Проміжний результат |
| Кількість цифр у поточному n | цілий | counter | Проміжний результат |
| Значення суми для поточного n | цілий | sum | Проміжний результат |
| Рахівник циклу | цілий | i | Індекс ітерації арифметичного циклу |

3)Програмні специфікації напишемо у псевдокоді та графічній формі у вигляді блок-схеми.

*Крок 1:* Визначимо основні дії.

*Крок 2:* Деталізуємо дію зміни знаку n в залежності від значення вводу користувача та оголошення змінних програми.

*Крок 3:* Деталізуємо зовнішній цикл програми.

*Крок 4:* Деталізуємо внутрішні цикли програми.

4)Псевдокод

*Крок 1*

**Початок**

-**ввід** n

- зміни знаку n в залежності від значення вводу користувача та оголошення змінних програми

- реалізація зовнішнього циклу програми

-реалізація внутрішніх циклів програми

**Кінець**

*Крок 2*

**Початок**

**-ввід** n

-**якщо** n<0

**то**:

n:=- n;

**все якщо**

оголошення copy, counter, sum;

- реалізація зовнішнього циклу програми

-реалізація внутрішніх циклів програми

**Кінець**

*Крок 3*

**Початок**

-**ввід** n

-**якщо** n<0

**то**:

n:=- n;

**все якщо**

**-оголошення** copy, counter, sum;

- **повторити поки** n>=10

copy:=n;

counter:=0;

sum:=0;

-реалізація внутрішніх циклів програми

n:=sum;

**все повторити**

**Кінець**

*Крок 4*

**Початок**

-**ввід** n

-**якщо** n<0

**то**:

n:=- n;

**все якщо**

**-оголошення** copy, counter, sum;

- **повторити поки** n>=10

copy:=n;

counter:=0;

sum:=0;

**повторити поки** copy>=1

copy:/=10;

increment counter;

**все повторити**

**для** і:=pow(10,counter-1) **до** 1 **якщо крок** і:/=10; **повторити**

add:=(n/i);

sum:+=add;

n:-=(add\*i);

**все повторити**

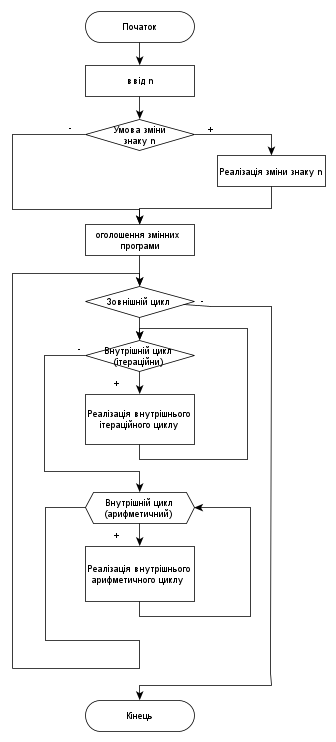
n:=sum;

**все повторити**

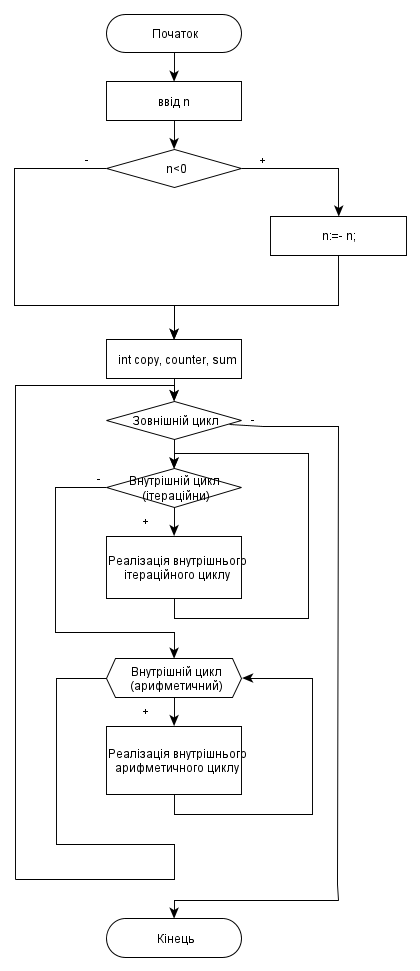
**Кінець**

**Блок-Схема**

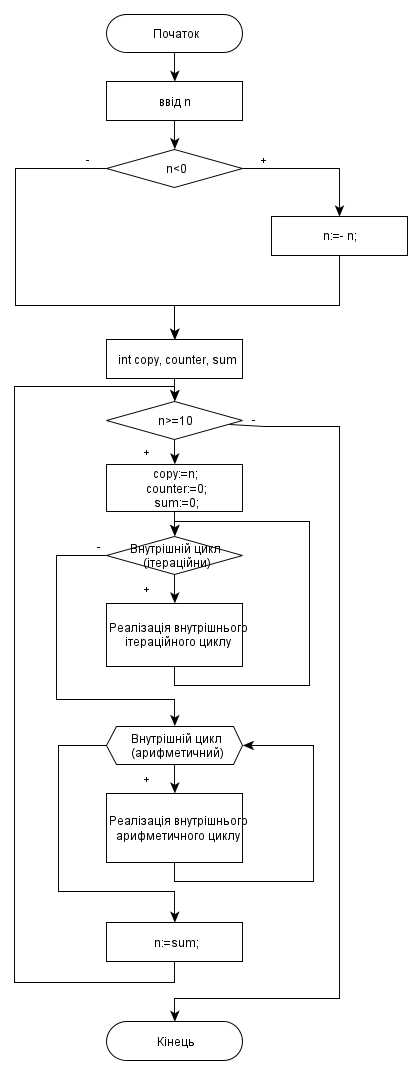
**Крок 1**

****

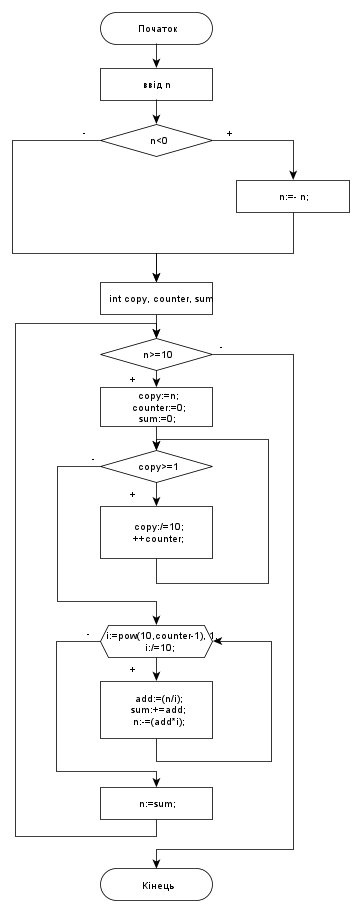
**Крок 2**

****

**Крок 3**

****

**Крок 4**

****

Перевіримо правильність алгоритму на довільних конкретних значеннях початкових даних:

|  |  |
| --- | --- |
| Блок | Дія |
|  | Початок |
| 1 | Введення n=29; |
| 2 | Перша ітерація зовнішнього циклу: n=11; |
| 3 | Друга ітерація зовнішнього циклу: n=2; |
|  | Кінець |

**Висновок:**

В ході виконання даної лабораторної роботи я навчився працювати з складними циклами, досліджував нелінійні програмні специфікації для визначення і обчислення виразів умова яких залежить від певних параметрів вводу, набув практичних навичок даних використання складних циклічних та розгалужених алгоритмів під час складання нелінійних програмних специфікацій. Навчився створювати блок-схеми алгоритмів що містять складні цикли та описувати програму за допомогою псевдокоду. Використовуючи отримані знання я успішно обрахував і відобразив на екрані значення цифрового кореня залежно від значення n, на компільованій мові С++ та інтерпретованій мові python.