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 № 6 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження лінійного пошуку в послідовностях »

Варіант 17

Виконав студент ІП-14 Макаревич Владислав Анатолійович

(шифр, прізвище, ім'я, по батькові)

Перевірив Мартинова Оксана Петрівна

( прізвище, ім'я, по батькові)

Київ 202 1

**Лабораторна робота 7**

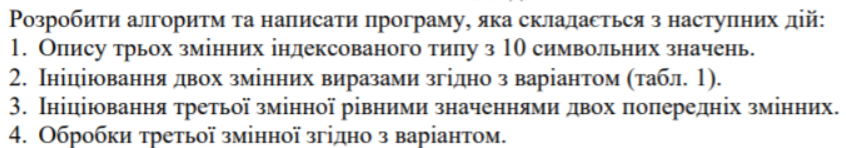
**Варіант 17**

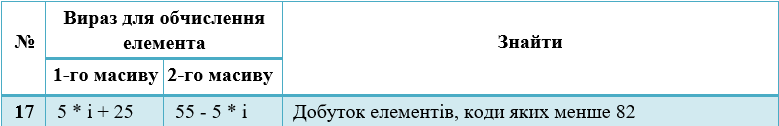
**Тема роботи:** Дослідження лінійного пошуку в послідовностях

**Мета роботи:** дослідити методи послідовного пошуку у впорядкованих і невпорядкованих послідовностях та набути практичних навичок їх використання під час складання програмних специфікацій.

**Хід роботи**

**Завдання**

****



**Постановка задачі**

Заданий алгоритм повинен:

1. Створити 2 змінні індексованого типу з 10 символьних значень.
2. Ініціювати перші 2 змінні за формулами 5𝑖 + 25 та 55 − 5𝑖, де 𝑖 .
3. Створення третьої змінної та ініціалізація її спільними значеннями двох попередніх.
4. Знайти добуток елементів третьої змінної, коди яких менше 82.

**Математична модель**

|  |  |  |  |
| --- | --- | --- | --- |
| **Номер** | **Ім’я** | **Тип** | **Призначення** |
| **1** | **sizeArray** | **Цілий** | **Початкове дане(за умовою)** |
| **2** | **arr1** | **Символьний** | **Проміжна змінна** |
| **3** | **arr2** | **Символьний** | **Проміжна змінна** |
| **4** | **arr3** | **Символьний** | **Вихідне дане** |

Крок 1: Визначити основні дії

Крок 2: Ініціювання першої змінної виразом згідно варіанту, заповнюємо перший масив.

Крок 3: Ініціювання другої змінної виразом згідно варіанту, заповнюємо другий масив.

Крок 4: Ініціювання третьої змінної рівними значеннями двох попередніх змінних

Крок 5: Обробка третьої змінної добуток елементів, коди яких менше 82

Крок 6: Виведення

Псевдокод

**Крок 1**

**Початок**

***Ініціювання першої змінної***

***Ініціювання другої змінної***

***Ініціювання третьої змінної***

***Обробка третьої змінної***

Виведення

**Кінець**

**Крок 2**

Початок

sizeArray=10

counter =0

Функція FillArray1(char arr1[])

для i = 0; поки i < sizeArray; i++)

повторити

arr1[i] = 5 \* i + 25;

все повторити

***Ініціювання другої змінної***

***Ініціювання третьої змінної***

***Обробка третьої змінної***

Виведення

**Кінець**

**Крок 3**

Початок

sizeArray=10

counter=0

Функція FillArray1(char arr1[])

для i = 0; поки i < sizeArray; i++)

повторити

arr1[i] = 5 \* i + 25;

все повторити

Функція FillArray2(char arr2[]

для i = 0; поки i < sizeArray; i++)

повторити

arr2[i] = 55 - 5 \* i;

все повторити

***Ініціювання третьої змінної***

***Обробка третьої змінної***

Виведення

**Кінець**

**Крок 4**

Початок

sizeArray=10

counter =0

Функція FillArray1(char arr1[])

для i = 0; поки i < sizeArray; i++)

повторити

arr1[i] = 5 \* i + 25;

все повторити

Функція FillArray2(char arr2[]

для i = 0; поки i < sizeArray; i++)

повторити

arr2[i] = 55 - 5 \* i;

все повторити

Функція FillArray3(char arr1[], char arr2[], char arr3[])

для i = 0; поки i < sizeArray; i++)

повторити

для j = 0; поки j < sizeArray; j++)

повторити

якщо arr1[i] == arr2[j]

то arr3[counter] = arr1[i]

counter=counter+1

все якщо

все повторити

все повторити

***Обробка третьої змінної***

Виведення

**Крок 5**

Початок

sizeArray=10

counter =0

Функція FillArray1(char arr1[])

для i = 0; поки i < sizeArray; i++)

повторити

arr1[i] = 5 \* i + 25;

все повторити

Функція FillArray2(char arr2[]

для i = 0; поки i < sizeArray; i++)

повторити

arr2[i] = 55 - 5 \* i;

все повторити

Функція FillArray3(char arr1[], char arr2[], char arr3[])

для i = 0; поки i < sizeArray; i++)

повторити

для j = 0; поки j < sizeArray; j++)

повторити

якщо arr1[i] == arr2[j]

то arr3[counter] = arr1[i]

counter=counter+1

все якщо

все повторити

все повторити

Функція multiplieArray(char arr3[])

для i = 0; поки i < sizeArray; i++)

повторити

якщо arr3[i] < 82

то multiplie \*= int(arr3[i])

все якщо

все повторити

return multiplie

Виведення

**Кінець**

**Крок 6**

Початок

sizeArray=10

counter =0

Функція FillArray1(char arr1[])

для i = 0; поки i < sizeArray; i++)

повторити

arr1[i] = 5 \* i + 25;

все повторити

Функція FillArray2(char arr2[]

для i = 0; поки i < sizeArray; i++)

повторити

arr2[i] = 55 - 5 \* i;

все повторити

Функція FillArray3(char arr1[], char arr2[], char arr3[])

для i = 0; поки i < sizeArray; i++)

повторити

для j = 0; поки j < sizeArray; j++)

повторити

якщо arr1[i] == arr2[j]

то arr3[counter] = arr1[i]

counter=counter+1

все якщо

все повторити

все повторити

Функція multiplieArray(char arr3[])

для i = 0; поки i < sizeArray; i++)

повторити

якщо arr3[i] < 82

то multiplie \*= int(arr3[i])

все якщо

все повторити

return multiplie

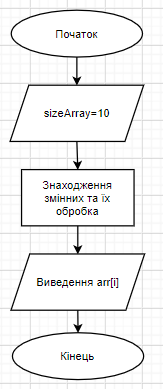
Функція ShowArray(char arr[])

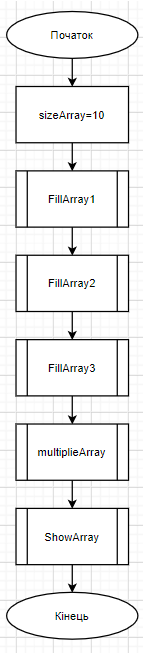
для i = 0; поки i < sizeArray; i++)

Виведення arr[]

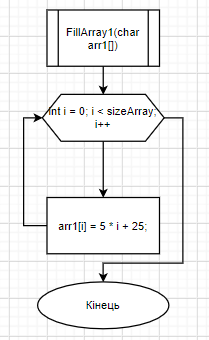
**Кінець**

**Блок-схема**

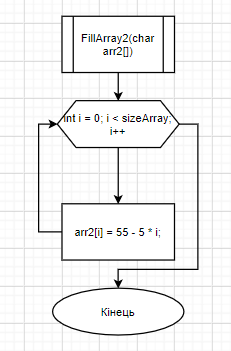
Крок 1



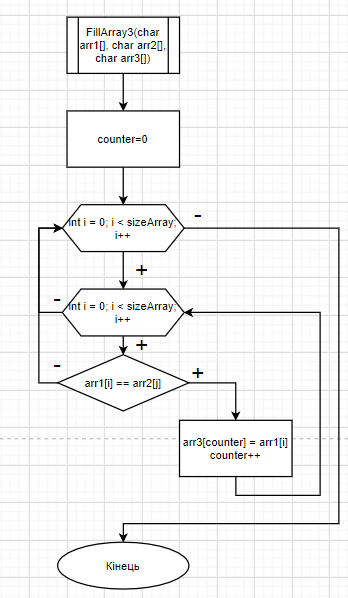
Крок 2



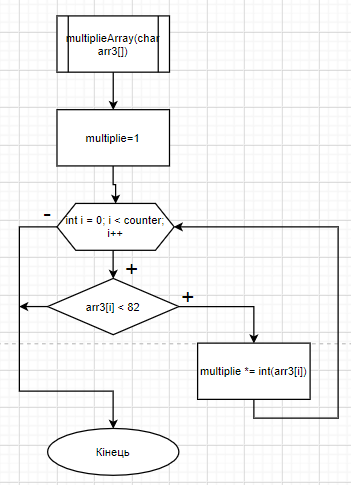
Крок 3



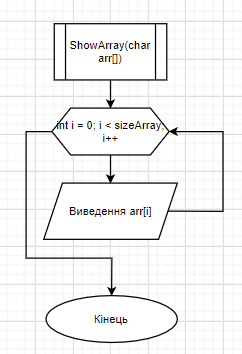
Крок 4



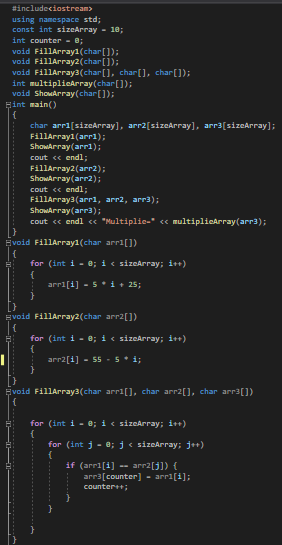
Крок 5

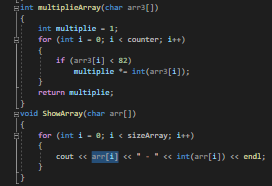


Крок 6

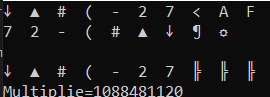


Код програми





Результат



Висновок: Протягом виконання цієї лабораторної роботи я набув навичок використання методів послідовного пошуку у невпорядкованих послідовностях та практичних навичок їх використання під час складання програмних специфікацій. Маючи формули задання елементів двох одновимірних символьних масивів, я склав програму яка успішно знаходить масив, який складається з спільних елементів попередніх масивів. Для отримання результату потрібно було використати математичну модель, псевдокод, а також блок-схему. Робота написана на с++.