

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 № 7 з дисципліни
«Алгоритми та структури даних-1.
Основи алгоритмізації»

« Дослідження лінійного пошуку в
послідовностях »

Варіант 34

Виконав студент Щербацький Антон ПІ-14
(шифр, прізвище, ім'я, по батькові)

Перевірив Доцент кафедри іпІ Мартинова О. П.
(прізвище, ім'я, по батькові)

Київ 2021

Лабораторна робота №7

Дослідження лінійного пошуку в послідовностях

Мета – дослідити методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях та набути практичних навичок їх використання під час складання програмних специфікацій.

Варіант - 34

Постановка задачі:

Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опису трьох змінних індексованого типу з 10 символічних значень.
2. Ініціювання двох змінних виразами згідно з варіантом (табл. 1).
3. Ініціювання третьої змінної рівними значеннями двох попередніх змінних.
4. Обробки третьої змінної згідно з варіантом.

34	$44 + 2 * i$	$55 - 2 * i$	Кількість елементів, коди яких діляться на 3
----	--------------	--------------	----------------------------------------------

Для ініціювання двох масивів використаємо арифметичний цикл for. Ітератор циклу буде відповідати індексу елементу масиву. Для заповнення третього масиву використаємо також вкладений цикл for, за допомогою зовнішнього циклу буде здійснено перебір елементів першого масиву, порівняння із елементами другого масиву буде застосовано вкладений цикл. Для обробки третього створеного масиву також використаємо цикл for та створимо змінну для збереження результату.

Математична модель:

Змінна	Тип	Ім'я	Призначення
Перший масив	Символьний(індексований)	Arr1	Проміжні дані
Другий масив	Символьний(індексований)	Arr2	Проміжні дані
Третій масив	Символьний(індексований)	Arr3	Проміжні дані
К-ть елементів які діляться на 3	Натуральний	quantity	Результат

1. Визначимо основні дії.
2. Ініціюємо початкові змінні(масиви)

3. Деталізуємо заповнення 2 масивів
4. Деталізуємо виведення масивів
5. Деталізуємо заповнення третього масиву
6. Деталізуємо знаходження значення змінної quantity

Псевдокод

Крок 1

Початок

Ініціалізація arr1, arr2, arr3, quantity

Заповнення arr 1, arr2

Виведення arr 1, arr2

Заповнення arr3

Знаходження змінної quantity

Виведення quantity

Кінець

Крок 2

Початок

Arr1[10], arr2[10], arr[10], quantity = 0;

Заповнення arr 1, arr2

Виведення arr 1, arr2

Заповнення arr3

Знаходження змінної quantity

Виведення quantity

Кінець

Крок 3

Початок

Arr1[10], arr2[10], arr[10], quantity = 0;

Повторити

Для I від 0 до 10, $I = I + 1$

$Arr1[i] = 44 + 2 * i$

$Arr2[i] = 55 - 2 * i$

Все повторити

Виведення arr 1, arr2

Заповнення arr3

Знаходження змінної quantity

Виведення quantity

Кінець

Крок 4

Початок

Arr1[10], arr2[10], arr[10], quantity = 0;

Повторити

Для I від 0 до 10, $I = I + 1$

$Arr1[i] = 44 + 2 * i$

$Arr2[i] = 55 - 2 * i$

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Виведення arr1[i]

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Виведення arr2[i]

Все повторити

Деталізуємо заповнення третього масиву

Знаходження змінної quantity

Виведення quantity

Кінець

Крок 5

Початок

Arr1[10], arr2[10], arr[10], quantity = 0;

Повторити

Для I від 0 до 10, $I = I + 1$

$Arr1[i] = 44 + 2 * i$

$Arr2[i] = 55 - 2 * i$

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Виведення arr1[i]

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Виведення arr2[i]

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Повторити

Для j від 0 до 10, $j = j + 1$

Якщо arr1[i] = arr2[i]

То

Arr3[i] = arr1[i]

Інакше

Arr3[i] = '#'

Все якщо

Все повторити

Все повторити

Деталізуємо знаходження значення змінної quantity

Виведення quantity

Кінець

Крок 6

Початок

Arr1[10], arr2[10], arr[10], quantity = 0;

Повторити

Для I від 0 до 10, $I = I + 1$

Arr1[i] = $44 + 2 * i$

Arr2[i] = $55 - 2 * i$

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Виведення arr1[i]

Все повторити

Повторити

Для I від 0 до 10, $I = I + 1$

Виведення arr2[i]

Все повторити

Повторити

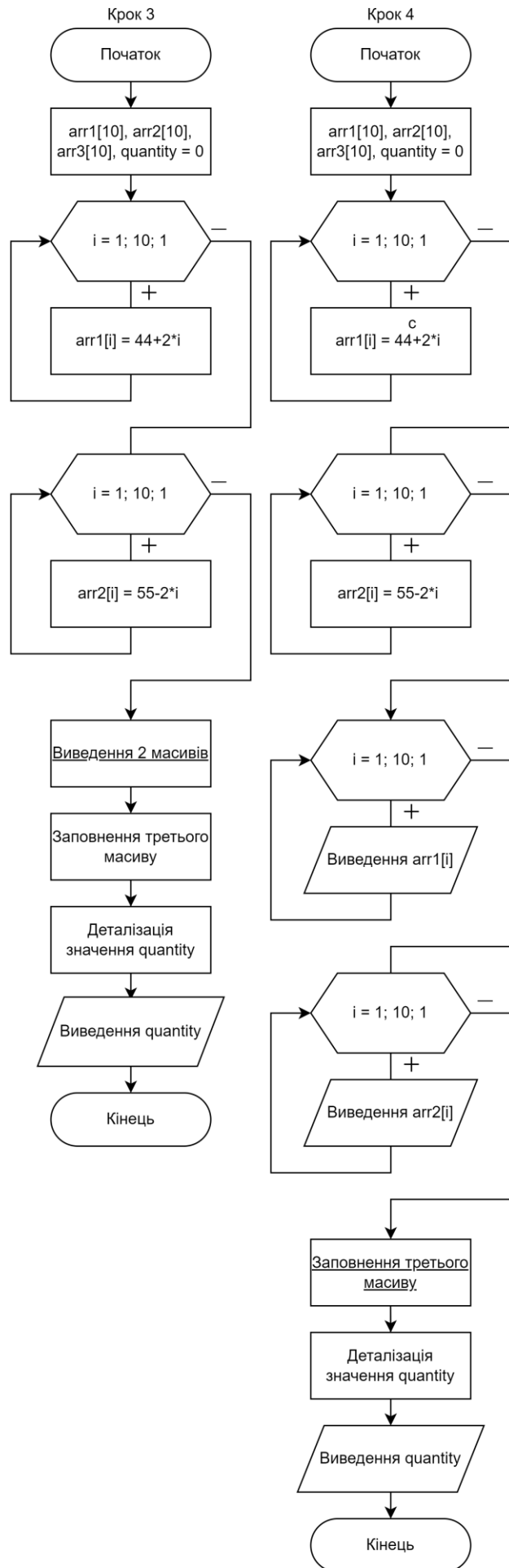
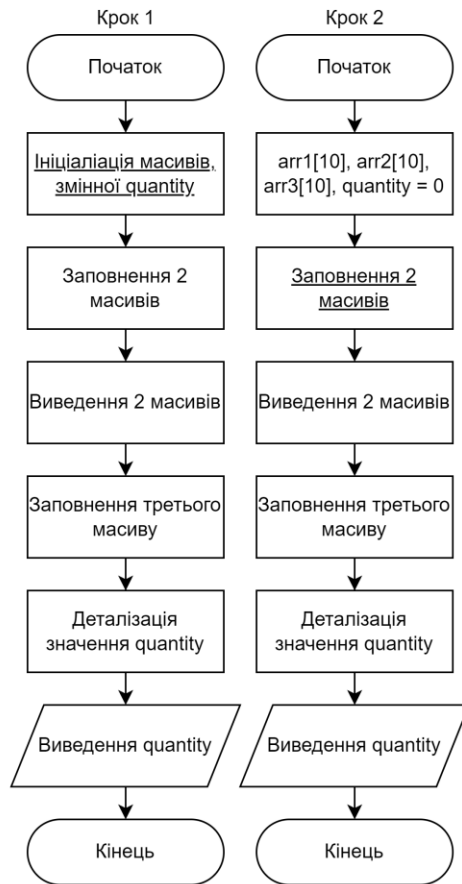
Для I від 0 до 10, $I = I + 1$

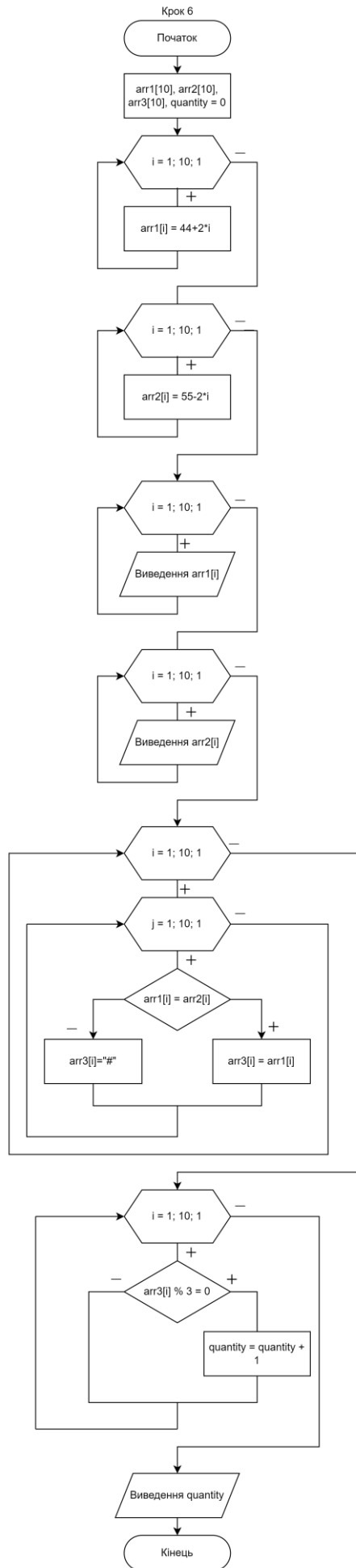
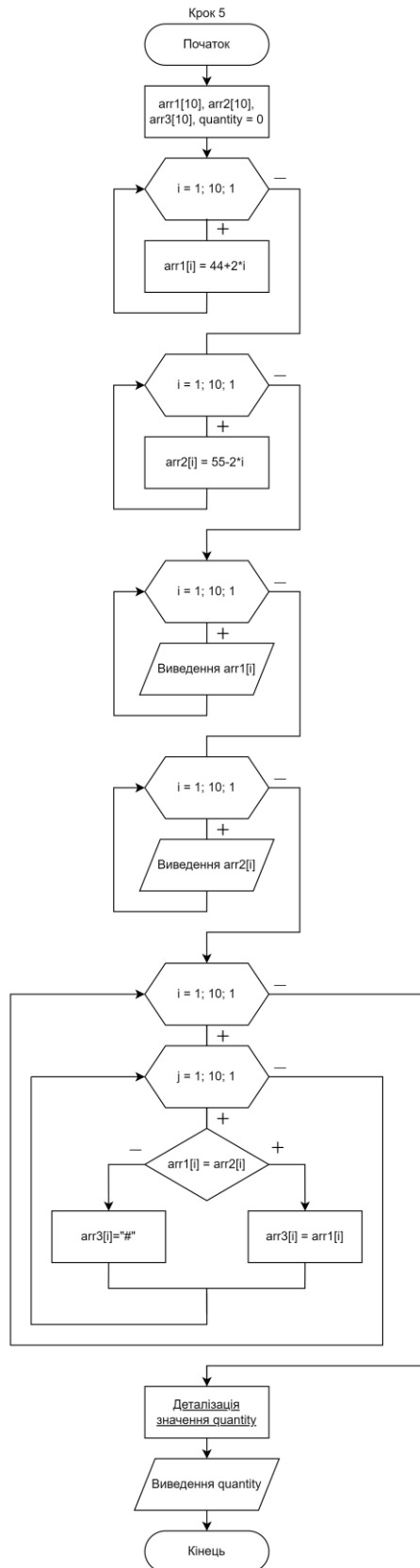
Повторити

Для j від 0 до 10, $j = j + 1$

```
        Якщо arr1[i] = arr2[i]
            То
                Arr3[i] = arr1[i]
        Інакше
            Arr3[i] = '#'
    Все якщо
Все повторити
Все повторити
Повторити
    Для I від 0 до 10, I = I + 1
        Якщо arr3[i] % 3 = 0
            То
                Quantity = quantity + 1
        Все якщо
Все повторити
Виведення quantity
Кінець

Блок-схема
```





Код програми(C++)

```
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  int main()
5  {
6      char arr1[10]{}, arr2[10]{}, arr3[10]{};
7      int quantity = 0;
8      for(int i = 0; i < 10; i++){
9          arr1[i] = 44 + 2 * i;
10         arr2[i] = 55 - 2 * i;
11     }
12     cout << "arr1: ";
13     for(int i = 0; i < 10; i++){
14         cout << setw(4) << arr1[i];
15     }
16     cout << endl;
17     cout << "arr2: ";
18     for(int i = 0; i < 10; i++){
19         cout << setw(4) << arr2[i];
20     }
21     for(int i = 0; i < 10; i++){
22         for(int j = 0; j < 10; j++){
23             if(arr1[i] == arr2[j])
24                 arr3[i] = arr1[i];
25             else
26                 arr3[i] = '#';
27         }
28     }
29     cout << endl;
30     cout << "arr3: ";
31     for(int i = 0; i < 10; i++){
32         cout << setw(4) << arr3[i];
33     }
34     for(int i = 0; i < 10; i++){
35         if(arr3[i] % 3 == 0){
36             quantity += 1;
37         }
38     }
39     cout << endl;
40     cout << "Codes which divides by 3: " << quantity << endl;
41 }
```

```

arr1:    ,    .    0    2    4    6    8    :    <    >
arr2:    7    5    3    1    /    -    +    )    '    %
arr3:    #    #    #    #    #    #    #    #    #    #
Codes which divides by 3: 0

Process finished with exit code 0

```

Тестування програми

Крок	Дія
1	$Arr1[i] = 44 + 2 * i$
2	$Arr2[i] = 55 - 2 * i$
3	$Arr1[10] = , . 0 2 4 6 8 : < >$
4	$Arr2[10] = 7 5 3 1 / - +) ' \%$
5	$\{,\} = \{7\} = \text{false}, arr3[1] = \#$
6	$\{.\} = \{5\} = \text{false}, arr3[2] = \#$
7	$\{0\} = \{3\} = \text{false}, arr3[3] = \#$
8	$\{2\} = \{1\} = \text{false}, arr3[4] = \#$
9	$\{4\} = \{/ \} = \text{false}, arr3[5] = \#$
10	$\{6\} = \{- \} = \text{false}, arr3[6] = \#$
11	$\{8\} = \{+ \} = \text{false}, arr3[7] = \#$
12	$\{:\} = \{ \} = \text{false}, arr3[8] = \#$
13	$\{< \} = \{ \} = \text{false}, arr3[9] = \#$
14	$\{> \} = \{ \% \} = \text{false}, arr3[10] = \#$
15	$Arr3[10] = \# \# \# \# \# \# \# \# \# \#$
16	Спільних елементів немає

Висновок

На цій лабораторній роботі я дослідив алгоритми лінійного пошуку у послідовностях. Для розв'язку даної задачі я використав арифметичний цикл, для знаходження спільних елементів також було використано декілька вкладених арифметичних циклів.

Було побудовано математичну модель, псевдокод, блок-схему, код програми та тестування. На прикладі мого завдання можна побачити, що спільних елементів у двох масивах немає, тому елементів які діляться націло на 3 із третього масиву також немає.