

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__30__

Виконав студент _____ ІП-15 Розін Олексій Іванович _____
(шифр, прізвище, ім'я, по батькові)

Перевірив _____ Вечерковська Анастасія Сергіївна _____
(прізвище, ім'я, по батькові)

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.

Індивідуальне завдання**Варіант 30****Постановка задачі**

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

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

Вираз для обчислення елемента		Знайти
1-го масиву	2-го масиву	
$43 - i$	$37 + i$	Добуток елементів, коди яких більше 40

Математична модель

Змінна	Тип	Ім'я	Призначення
Перший масив	Символьний	arr1	Початкові дані
Другий масив	Символьний	arr2	Початкові дані
Третій масив	Символьний	arr3	Проміжні дані
Лічильник зовнішнього циклу	Цілий	i	Проміжні дані
Лічильник внутрішнього циклу	Цілий	k	Проміжні дані
Результат	Цілий	res	Вихідні дані

Оголошуємо змінні arr1, arr2, arr3 індексованого типу з 10 символьних значень. Ініціалізуємо змінну res (res = 1). За допомогою арифметичного з лічильником i = 0; умовою i < 10 та кроком i += 1 ініціалізуємо змінні arr1, arr2, arr3 (arr1[i] = 43 - I; arr2[i] = 37 + I; arr3[i] = 0). Щоб ініціювати третю змінну рівними значеннями двох попередніх змінних використаємо вкладений арифметичний цикл. У тілі зовнішнього циклу з лічильником i = 0; умовою i < 10 та кроком i += 1 завдамо вкладений арифметичний цикл з лічильником k = 0; умовою k < 10 та кроком k += 1. У тілі внутрішнього циклу за допомогою умовного оператора ініціалізуємо змінну arr3 (якщо arr1[i] == arr2[k], то arr3[i] = arr1[i]). Далі за допомогою арифметичного циклу та умовного оператора обчислюємо значення res. В тілі арифметичного циклу з лічильником i = 0; умовою i < 10 та кроком i += 1 використовуємо умовний оператор (якщо arr3[i] > 0, то res *= arr3[i]). В кінці виводимо значення змінної res.

Розв'язання

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.

Крок 1. Визначимо основні дії.

Крок 2. Ініціалізація arr1, arr2, arr3, res.

Крок 3. Ініціалізація arr3.

Крок 4. Обчислення res.

Крок 5. Виведення res.

Псевдокод

Крок 1

початок

для i від 0 до 9 повторити

$arr1[i] = 43 - i$

$arr2[i] = 37 + i$

$arr3[i] = 0$

все повторити

Ініціалізація arr3

Обчислення res

Виведення res

кінець

Крок 2

початок

для i від 0 до 9 повторити

$arr1[i] = 43 - i$

$arr2[i] = 37 + i$

$arr3[i] = 0$

все повторити

для i від 0 до 9 повторити

для k від 0 до 9 повторити

якщо $\text{arr1}[i] == \text{arr2}[k]$

то $\text{arr3}[i] = \text{arr1}[i]$

все якщо

все повторити

все повторити

Обчислення res

Виведення res

кінець

Крок 3

початок

для i від 0 до 9 повторити

$\text{arr1}[i] = 43 - i$

$\text{arr2}[i] = 37 + i$

$\text{arr3}[i] = 0$

все повторити

для i від 0 до 9 повторити

для k від 0 до 9 повторити

якщо $\text{arr1}[i] == \text{arr2}[k]$

то $\text{arr3}[i] = \text{arr1}[i]$

все якщо

все повторити

все повторити

для i від 0 до 9 повторити

якщо $\text{arr3}[i] > 40$

to res *= arr3[i]

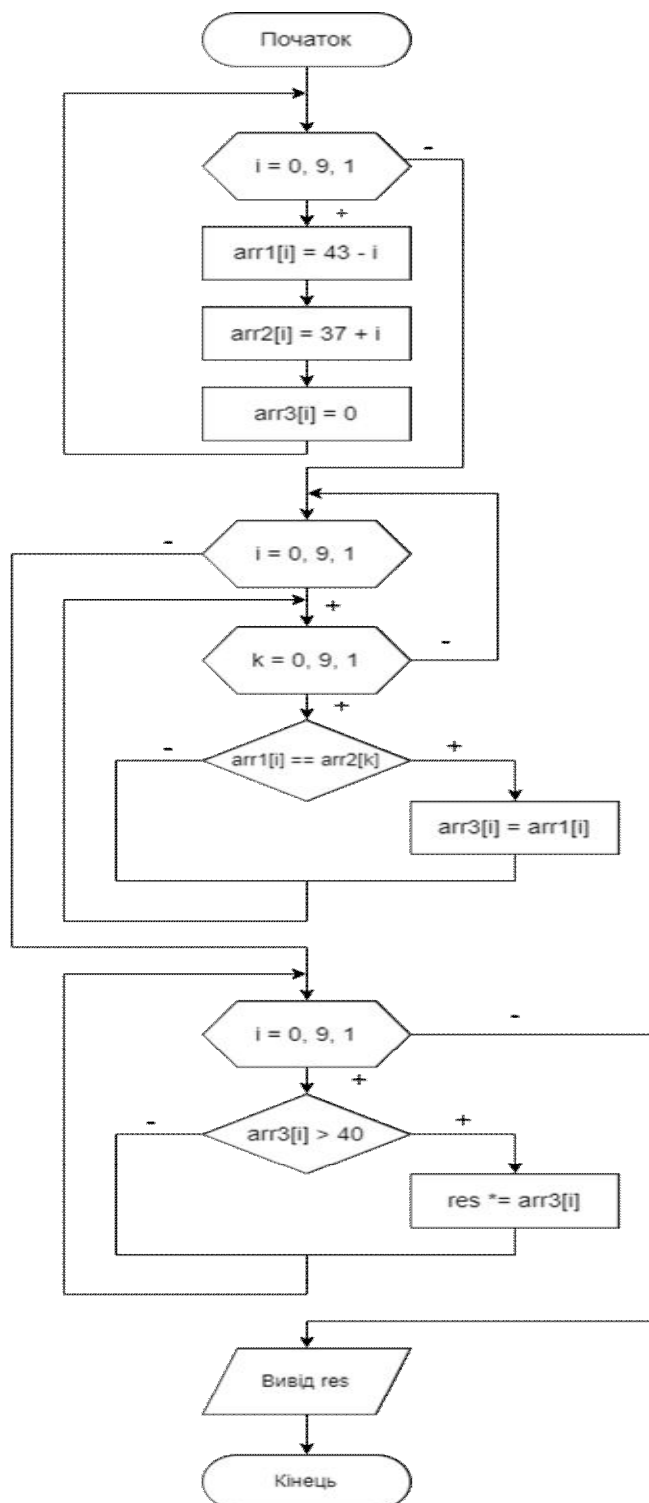
все якщо

все повторити

Виведення res

кінець

Блоксхема



Код програми

```
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  int main()
5  {
6      char arr1[10];
7      char arr2[10];
8      char arr3[10];
9      int res = 1;
10     for (int i = 0; i < 10; i++) {
11         arr1[i] = 43 - i;
12         arr2[i] = 37 + i;
13         arr3[i] = 0;
14     }
15     for (int i = 0; i < 10; i++) {
16         for (int k = 0; k < 10; k++) {
17             if (arr1[i] == arr2[k]) {
18                 arr3[i] = arr1[i];
19             }
20         }
21     }
22     for (int i = 0; i < 10; i++) {
23         if (arr3[i] > 40) {
24             res *= arr3[i];
25         }
26     }
27
28     // вивод результату
29     cout << "arr1  arr2  arr3\n-----\n";
30     for (int i = 0; i < 10; i++) {
31         cout << arr1[i] << setw(7) << arr2[i] << setw(7) << arr3[i] << endl;
32     }
33     cout << endl << "Result: " << res << endl;
34 }
```

Консоль отладки Microsoft Visual Studio

```
arr1  arr2  arr3
-----
+      %      +
*      &      *
)      '      )
(      (      (
'      )      '
&      *      &
%      +      %
$      ,
#      -
"      .

Result: 74046
```

Випробування

Блок	Дія
	Початок
1	Робота першого циклу (i = 0, 9, 1)
	arr1 = {+, *,), (, ', &, %, \$, #, ""}
	arr2 = {% , &, ' , (,) , * , + , , , - , . }
2	arr3 = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
	Робота другого циклу (i = 0, 9, 1)

2.1	Робота внутрішнього циклу ($k = 0, 9, 1$)
	$arr3 = \{+, *,), (, ', \&, \%\}$
3	Робота третього циклу
	$res = 74046$
4	Вивід res

Висновки

Ми дослідили методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях та набули практичних навичок їх використання під час складання програмних специфікацій. В результаті виконання лабораторної роботи ми отримали алгоритм знаходження масиву з елементами, які є однаковими значеннями двох даних нам масивів та знайшли добуток елементів знайденого масива, коди яких більші за 40.