

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 6

Виконав студент ІП-13 Вдовиченко Станіслав Юрійович
(шифр, прізвище, ім'я, по батькові)

Перевірів 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.

Варіант 6:

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: 1-ий масив = $73 - i$,
2-ий масив = $64 + 2 * i$.
3. Ініціювання третьої змінної рівними значеннями двох попередніх змінних.
4. Обробки третьої змінної: знаходження кількості елементів між максимальним та мінімальним елементами.

Постановка задачі: треба задати три масиви з розмірністю 10, два з них заповнити елементами за виразами для обчислення. Третій масив заповнити однаковими елементами двох попередніх масивів, вільні значення заповнити нулями. За допомогою арифметичних циклів знайти мінімальне і максимальне значення масиву та кількість елементів між цими значеннями.

Математична модель:

Змінна	Тип	Ім'я	Призначення
Перший масив	Символьний	array1	Проміжні дані
Другий масив	Символьний	array2	Проміжні дані
Третій масив	Символьний	array3	Проміжні дані
Максимум масиву	Символьний	max	Проміжні дані
Мінімум масиву	Символьний	min	Проміжні дані
К-сть елементів	Цілий	number	Вихідні дані

Таким чином математичне формулювання задачі зводиться до обчислення двох масивів розмірністю 10, де елементи знаходяться з використанням арифметичного циклу за формулою:

1-ий масив: $73 - i$,

2-ий масив: $64 + 2 * i$,

де i – порядковий номер елемента в масиві ($i \in [0,9]$).

Третій масив за допомогою арифметичного циклу спочатку заповнюємо нулями, а потім замінюємо нулі на однакові елементи першого і другого масивів(якщо такі елементи існують).

Щоб знайти максимальний елемент 3-го масиву, задаємо змінну $max = 0$ (тобто найменший елемент знакового типу в таблиці ASCII). За допомогою арифметичного циклу знаходимо найбільше значення, тобто:

$\text{max} = 0$, якщо ж i -ий елемент масиву більше за max , то $\text{max} = i$ -ий елемент.

Так перевіряємо кожен елемент масиву.

Щоб знайти мінімальний елемент масиву, задаємо змінну $\text{min} = 127$ (найбільше значення знакового типу за таблицею ASCII). За допомогою арифметичного циклу шукаємо найменше значення, тобто:

$\text{min} = 127$, якщо ж i -ий елемент масиву менше за min , то $\text{min} = i$ -ий елемент.

Щоб знайти кількість елементів між мінімальним і максимальним значенням, задаємо змінну $\text{amount} = 0$. За допомогою арифметичного циклу знаходимо k -сть, тобто:

якщо i -ий елемент масиву не дорівнює максимуму, мінімуму і нулю, то додаємо 1 до змінної amount .

Для запису даного алгоритму використаємо підпрограми:

- `firstArray`, `secondArray`, `thirdArray` типу `void` (які нічого не повертають), що будуть приймати як параметр покажчики на масиви та будуть заповнювати його за формулою.
- `maxOfArray`, `minOfArray` символьного типу (які повертають значення символьного типу), що будуть приймати як параметр покажчик на третій масив та будуть повертати відповідно максимальне та мінімальне значення.
- `numberOfSymbols` цілочисельного типу (яка повертає ціле число), що буде приймати як параметри максимальне, мінімальне значення та покажчик на 3 масив, а повертати буде кількість елементів між максимумом і мінімумом.

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

Крок 1. Визначимо основні дії.

Крок 2. Ініціювання масивів.

Крок 3. Заповнення 1-го масиву.

Крок 4. Заповнення 2-го масиву.

Крок 5. Заповнення 3-го масиву.

Крок 6. Знаходження максимального значення масиву.

Крок 7. Знаходження мінімального значення масиву.

Крок 8. Знаходження кількості елементів між мінімальним та максимальним значеннями.

Псевдокод

Функція

`firstArray(*array1[])`

повторити від 0 до 10 з кроком 1

`*array1[i] = 73 - i`

все повторити

Все функція

Функція

secondArray(*array2[])

повторити від 0 до 10 з кроком 1

***array2[i] = 64 + 2 * i**

все повторити

Все функція

Функція

thirdArray(*array1, *array2, *array3)

повторити від 0 до 10 з кроком 1

***array3[i] = 0**

все повторити

повторити від 0 до 10 з кроком 1

повторити від 0 до 10 з кроком 1

якщо (*array1[i] = *array2[j])

***array3[i] = *array2[j]**

все якщо

все повторити

все повторити

Все функція

Функція

maxOfArray(*array3)

maximum = 0

повторити від 0 до 10 з кроком 1

якщо (maximum < *array3[i])

maximum = *array3[i]

все якщо

все повторити

return maximum

Все функція

Функція

minOfArray(*array3[])

minimum = 127

повторити від 0 до 10 з кроком 1

якщо (minimum > *array3[i] && *array3[i] != 0)

minimum = *array3[i]

все якщо

все повторити

return minimum

Все функція

Функція

numberOfSymbols(min, max, *array3)

quantity = 0

повторити від 0 до 10 з кроком 1

якщо (*array3[i] != max && *array3[i] != min && *array3[i] != 0)

quantity += 1

все якщо

все повторити

return quantity

Все функція

Початок

array1[10]

array2[10]

array3[10]

firstArray(array1)

secondArray(array2)

thirdArray(array1,array2,array3)

max = maxOfArray(array3)

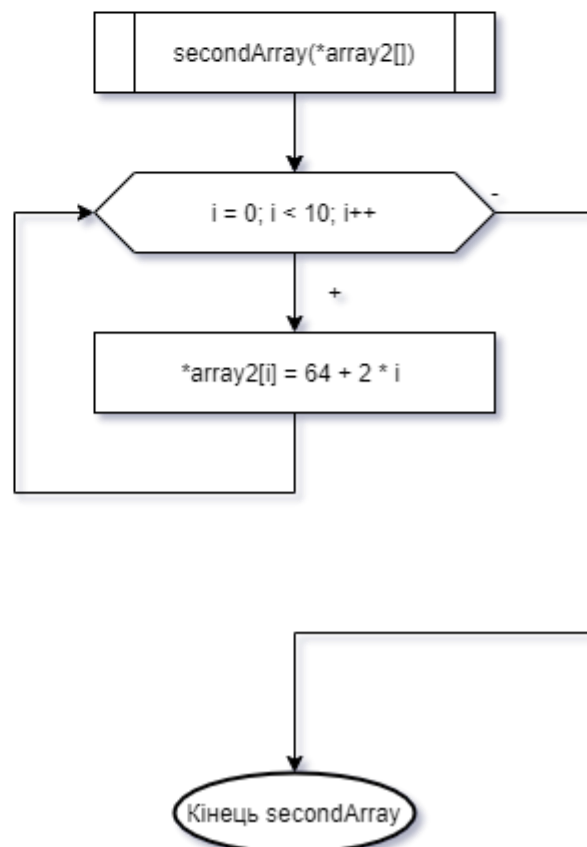
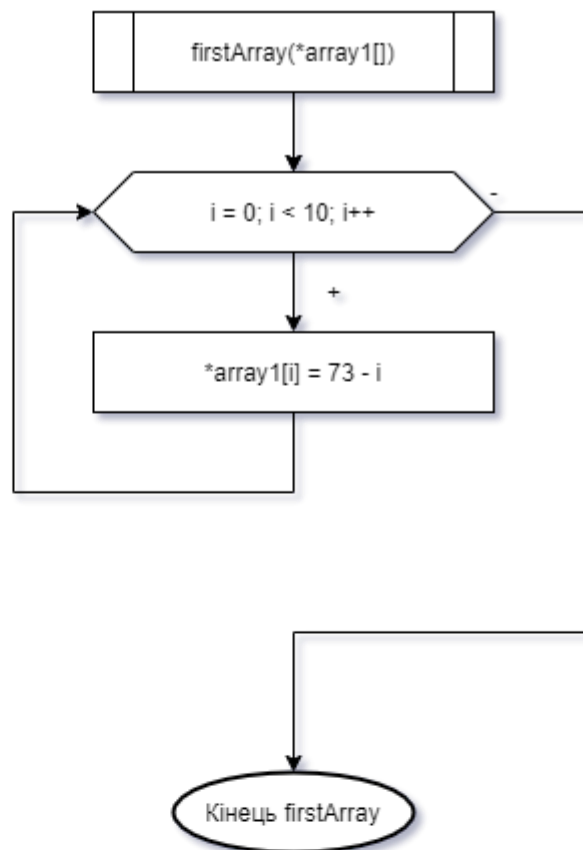
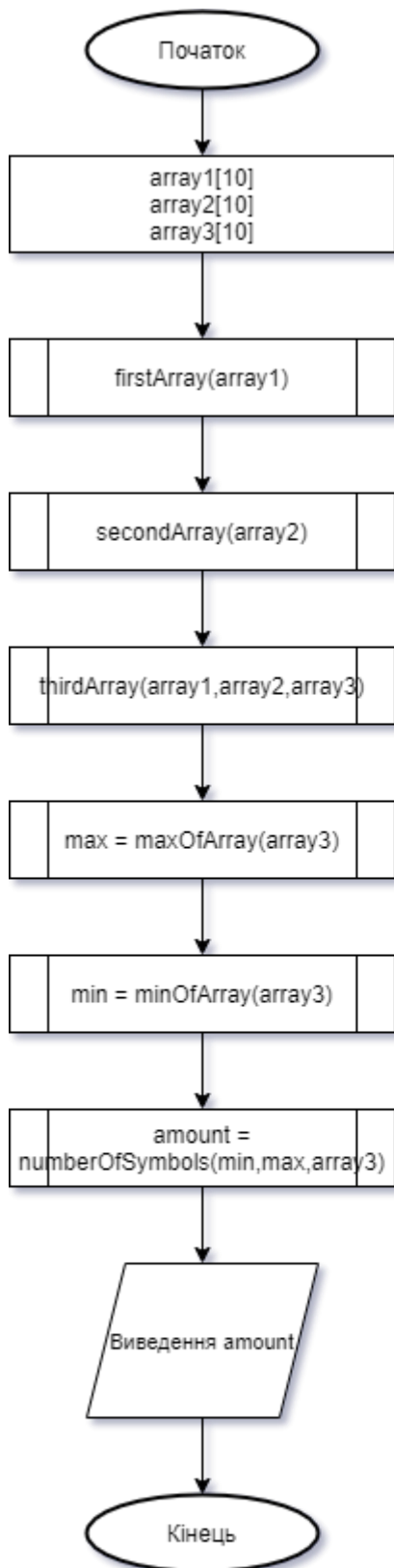
min = minOfArray(array3)

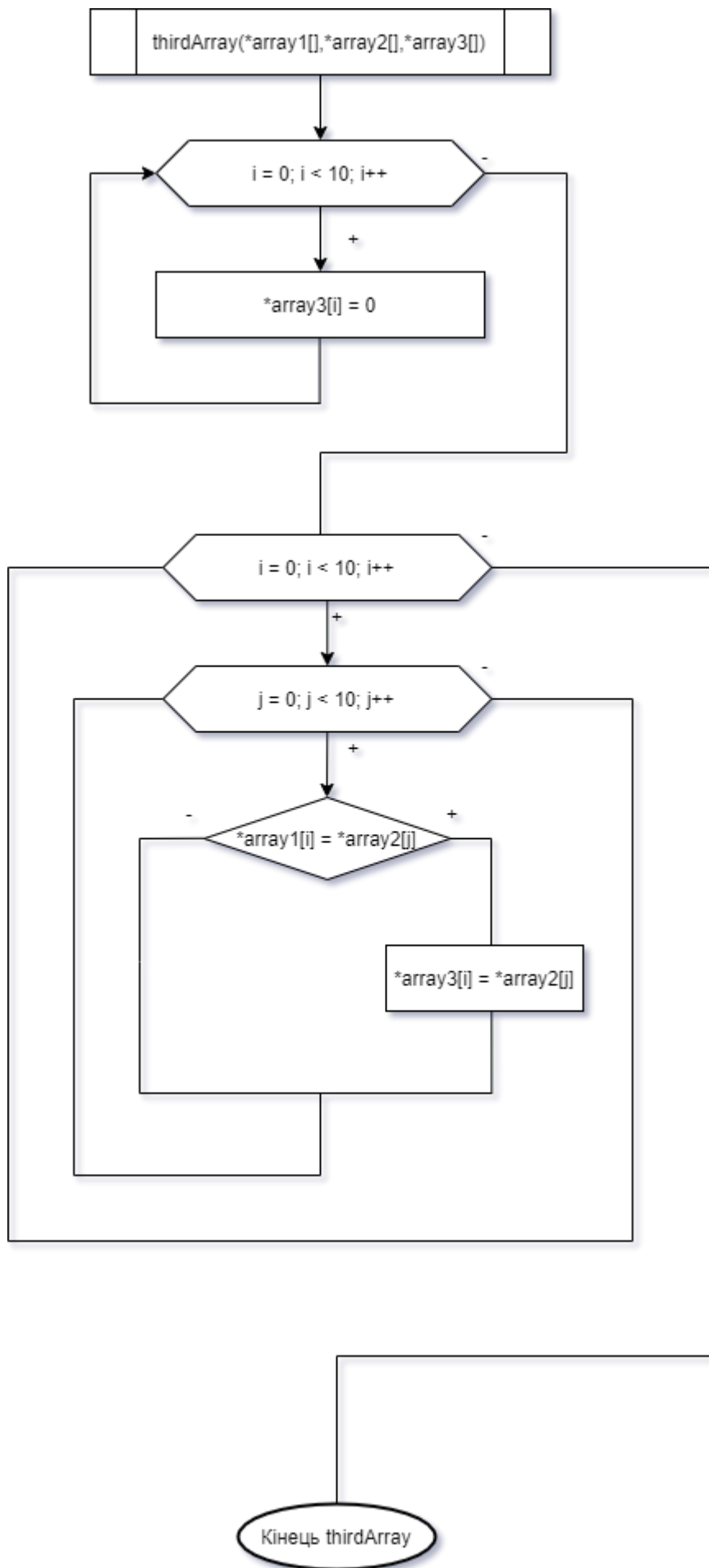
number = numberOfSymbols(min,max,array3)

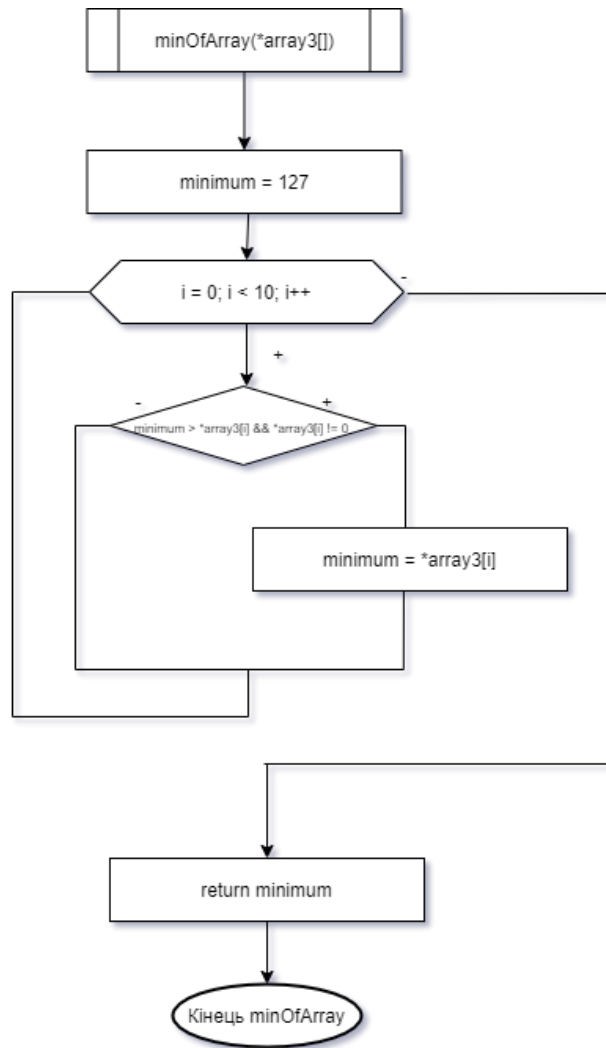
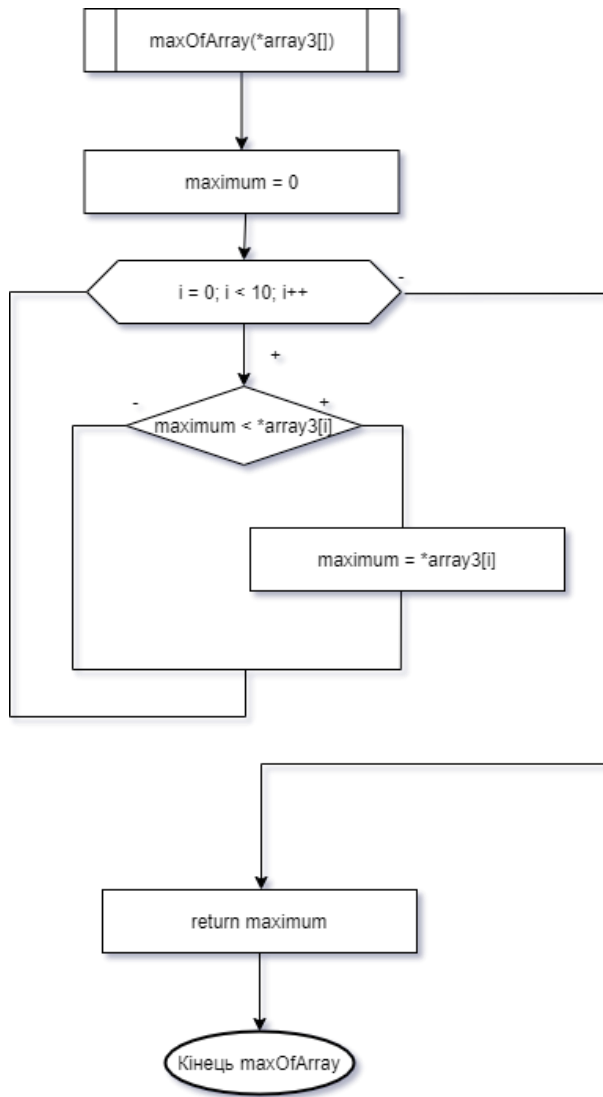
Виведення amount

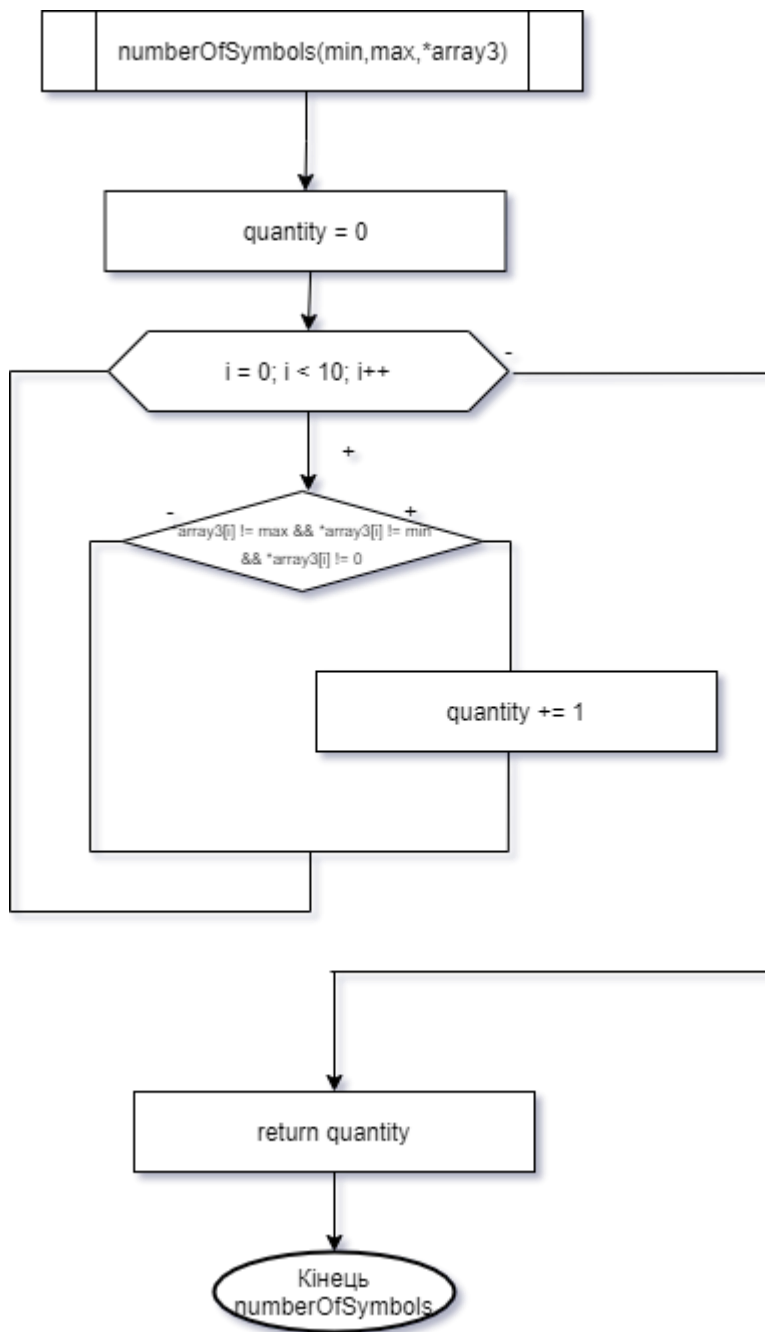
Кінець

Блок-схема









Код програми

// функція outArray написана для того, щоб користувач міг побачити етапи виконання програми, на алгоритм не впливає.

```
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  void outArray(char array[]);
6  void firstArray(char array1[]);
7  void secondArray(char array2[]);
8  void thirdArray(char array1[], char array2[], char array3[]);
9  char maxOfArray(char array3[]);
10 char minOfArray(char array3[]);
11 int  numberOfSymbols(char max, char min, char array3[]);
12
13 int main() {
14     int  number;
15     char max;
16     char min;
17     char array1[10]; // ініціалізація масивів
18     char array2[10];
19     char array3[10];
20
21     firstArray(array1); // обчислення 1-го масиву
22     secondArray(array2); // обчислення 2-го масиву
23     thirdArray(array1,array2,array3); // обчислення 3-го масиву
24     max = maxOfArray(array3); // обчислення максимального значення 3-го масиву
25     min = minOfArray(array3); // обчислення мінімального значення 3-го масиву
26     number = numberOfSymbols(max, min, array3); // обчислення к-сті елементів між мінімумом і максимумом
27
28     cout << "First array: "; outArray(array1);
29     cout << "Second array: "; outArray(array2);
30     cout << "Third array: "; outArray(array3);
31     cout << "Max of Array: " << max << endl;
32     cout << "Min of Array: " << min << endl;
33     cout << "-----" << endl;
34     cout << "The number of symbols between min and max: " << number << endl;
35     cout << "-----" << endl;
36     return 0;
37 }
```

```

38 void outArray(char array[]){
39     for (int i=0; i < 10; i++){
40         cout << setw(3) << array[i];
41     }
42     cout << endl;
43 }
44 void firstArray(char array1[]) {
45     for(int i = 0; i < 10; i++) {
46         array1[i] = 73 - i;
47     }
48 }
49 void secondArray(char array2[]) {
50     for (int i = 0; i < 10; i++) {
51         array2[i] = 64 + 2 * i;
52     }
53 }
54 void thirdArray(char array1[], char array2[], char array3[]){
55     for (int i = 0; i < 10; i++) {
56         array3[i] = 0;
57     }
58     for(int i = 0; i < 10; i++){
59         for( int j = 0; j < 10; j++){
60             if(array1[i] == array2[j]){
61                 array3[i] = array2[j];
62             }
63         }
64     }
65 }
66 char maxOfArray(char array3[]){
67     char maximum = 0;
68     for(int i = 0; i < 10; i++){
69         if(maximum < array3[i]){
70             maximum = array3[i];
71         }
72     }
73     return maximum;
74 }

```

```

75 char minOfArray(char array3[]){
76     char minimum = 127;
77     for(int i = 0; i < 10; i++){
78         if(minimum > array3[i] && array3[i] != 0){
79             minimum = array3[i];
80         }
81     }
82     return minimum;
83 }
84 int numberOfSymbols(char max, char min, char array3[]){
85     int quantity = 0;
86     for (int i = 0; i < 10; i++){
87         if((array3[i] != max) && (array3[i] != min) && (array3[i] != 0)){
88             quantity += 1;
89         }
90     }
91     return quantity;
92 }

```

```

First array:  I  H  G  F  E  D  C  B  A  @
Second array: @  B  D  F  H  J  L  N  P  R
Third array:   H      F      D      B      @
Max of Array: H
Min of Array: @

```

```

-----
The number of symbols between min and max: 3
-----

```

```

Process finished with exit code 0

```

Тестування програми

Блок	Дія
	Початок
1	Заповнення першого масиву: $i = 0, \text{array1}[0] = 73 - 0 = 73 = \text{I}$ $i = 1, \text{array1}[1] = 73 - 1 = 72 = \text{H}$ $i = 2, \text{array1}[2] = 73 - 2 = 71 = \text{G}$ $i = 3, \text{array1}[3] = 73 - 3 = 70 = \text{F}$ $i = 4, \text{array1}[4] = 73 - 4 = 69 = \text{E}$ $i = 5, \text{array1}[5] = 73 - 5 = 68 = \text{D}$ $i = 6, \text{array1}[6] = 73 - 6 = 67 = \text{C}$ $i = 7, \text{array1}[7] = 73 - 7 = 66 = \text{B}$ $i = 8, \text{array1}[8] = 73 - 8 = 65 = \text{A}$ $i = 9, \text{array1}[9] = 73 - 9 = 64 = \text{@}$
2	Заповнення другого масиву $i = 0, \text{array2}[0] = 64 + 2 \cdot 0 = 64 = \text{@}$ $i = 1, \text{array2}[1] = 64 + 2 \cdot 1 = 66 = \text{B}$ $i = 2, \text{array2}[2] = 64 + 2 \cdot 2 = 68 = \text{D}$ $i = 3, \text{array2}[3] = 64 + 2 \cdot 3 = 70 = \text{F}$ $i = 4, \text{array2}[4] = 64 + 2 \cdot 4 = 72 = \text{H}$ $i = 5, \text{array2}[5] = 64 + 2 \cdot 5 = 74 = \text{J}$ $i = 6, \text{array2}[6] = 64 + 2 \cdot 6 = 76 = \text{L}$ $i = 7, \text{array2}[7] = 64 + 2 \cdot 7 = 78 = \text{N}$ $i = 8, \text{array2}[8] = 64 + 2 \cdot 8 = 80 = \text{P}$ $i = 9, \text{array2}[9] = 64 + 2 \cdot 9 = 82 = \text{R}$

3	Заповнення третього масиву: array1[1] = array2[4] = H array3[1] = H array1[3] = array2[3] = F array3[3] = F array1[5] = array2[2] = D array3[5] = D array1[7] = array2[1] = B array3[7] = B array1[9] = array2[0] = @ array3[9] = @
4	maximum = 0, <ul style="list-style-type: none"> • 0 < 72; maximum = 72 = H, • 70 < maximum • 68 < maximum • 66 < maximum • 64 < maximum max = H
5	minimum = 127 <ul style="list-style-type: none"> • minimum > 72, minimum = 72 = H ... • minimum > 64, minimum = 64 = @ min = @
6	quantity = 0 i = 3, quantity = 1 i = 5, quantity = 2 I = 7, quantity = 3 number = quantity
7	number = quantity = 3
	Кінець

Висновки

Під час виконання даної лабораторної роботи я дослідив методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях та набув практичних навичок їх використання під час складання програмних специфікацій. Розробив алгоритм для розв'язання поставленої задачі, побудував математичну модель, псевдокод, блок-схему. Написав код програми, протестував алгоритм.