

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 19
Виконав студент: ІІІ-15 Левченко Владислав В'ячеславович
(шифр, прізвище, ім'я, по батькові)
Перевірила: Вечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)
Київ 2021

19	$2 * i + 22$	$48 - 2 * i$	Суму кодів між максимальним та мінімальним елементами
----	--------------	--------------	---

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

Побудова математичної моделі

Постановка задачі. Для вирішення даної задачі спочатку створюємо три символні масиви $a[10]$, $b[10]$, $c[10]$. Створюємо цикли `for()` для обох масивів (a) і (b), змінну "i" використовуємо як показник у циклах. Використовуючи формули пошуку $2*i + 22(a)$ і $48 - 2*i(b)$, наповнюємо масиви елементами. Наступним кроком буде цикл з умовою, який наповнює масив (c) спільними для масивів (a) і (b) елементами. Коли

знайдено спільні елементи створюємо два цикли які шукають максимальний і мінімальний елементи масиву (c).

Таблиця імен змінних

Змінна	Тип	Ім'я	Призначення
Масив a	Символьний	a	Вхідне дане
Масив b	Символьний	b	Вхідне дане
Масив c	Символьний	c	Вхідне, проміжне дане
Показчик i	Ціле	i	Проміжне дане
Показчик j	Ціле	j	Проміжне дане
Показчик counter	Ціле	counter	Проміжне дане
Показчик w	Ціле	w	Проміжне дане
Показчик f	Ціле	f	Проміжне дане
Показчик k	Ціле	k	Проміжне дане
Показчик r	Ціле	r	
Змінна максимального елементу	Ціле	storageMax	Вихідне дане
Змінна мінімального елементу	Ціле	storageMin	Вихідне дане

Крок 1. Визначимо основні дії

Крок 2. Створення змінних і масивів

Крок 3. Створення циклів пошуку елементів для a і b

Крок 4. Створення циклу для пошуку елементів для масиву c

Крок 5. Створення циклів для пошуку максимального та мінімального елементу

Псевдокод

Крок 1

початок

Створення змінних і масивів

Створення циклів пошуку елементів для a і b

Створення циклу для пошуку елементів для масиву c

Створення циклів для пошуку максимального та мінімального елементу

кінець

Крок 2

початок

$a[10]$

$b[10]$

$c[10] = \{0\}$

storageMin

storageMax:= 0

Створення циклів пошуку елементів для a і b

Створення циклу для пошуку елементів для масиву c

Створення циклів для пошуку максимального та мінімального елементу

кінець

Крок 3

початок

$a[10]$

$b[10]$

$c[10] = \{0\}$

storageMin

storageMax:= 0

для j від 1 до 10

$b[j] := 48 - 2 * j$

для i від 1 до 10

$a[i] := 2 * i + 22$

Створення циклу для пошуку елементів для масиву c

Створення циклів для пошуку максимального та мінімального елементу

кінець

Крок 3

початок

$a[10]$

$b[10]$

$c[10] = \{0\}$

storageMin

storageMax := 0

для j від 0 до 10

$b[j] := 48 - 2 * j$

все повторити

для i від 0 до 10

$a[i] := 2 * i + 22$

все повторити

для k від 0 до 10

для f від 0 до 10

якщо $a[k] == b[f]$

то $c[k] = \text{int}(a[k])$

все повторити

для w від 0 до 10

виведення $c[w]$

Створення циклів для пошуку максимального та мінімального елементу

кінець

Крок 3

початок

a[10]

b[10]

c[10] = {0}

storageMin

storageMax:= 0

для j від 0 до 10

b[j]:= 48 – 2 * j

все повторити

для i від 0 до 10

a[i]:= 2 * i + 22

все повторити

для k від 0 до 10

для f від 0 до 10

якщо a[k] == b[f]

то c[k]:= int(a[k])

все якщо

все повторити

для w від 0 до 10

виведення c[w]

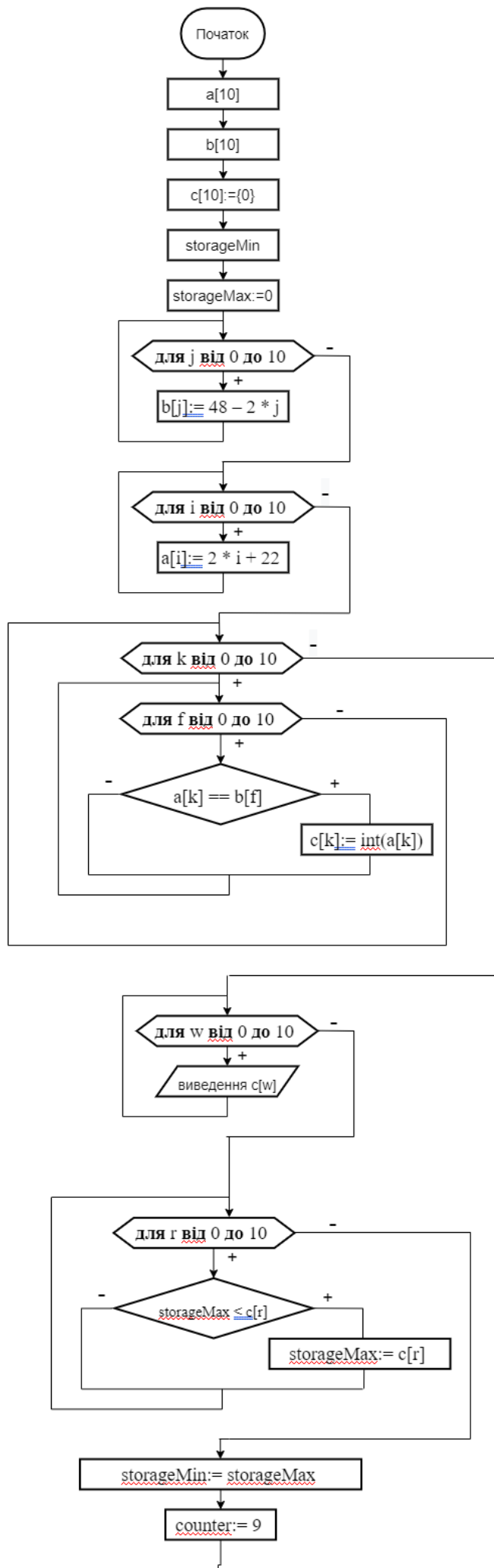
все повторити

для r від 0 до 10

якщо storageMax < c[r]

```
to storageMax:= c[r]
все якщо
все повторити
storageMin:= storageMax
counter:= 9
    повторити storageMin:= c[counter]
    counter--
    поки c[counter] > 0
все повторити
    виведення storageMax + storageMin
кінець
```

Блок схема



Код програми


```

#include <iostream>
#include <math.h>
#include <cstdio>
#include <cstdlib>

using namespace std;

int main()
{
    setlocale(LC_ALL, ".1251");

    int storageMin;
    int storageMax = 0;

    char a[10], b[10], c[10] = { 0 }; // Створюємо масиви і змінні

    std::cout << "The massive b: "; // Призначаємо значення у масив b
    for (int j = 0; j < 10; j++) {
        b[j] = 48 - 2 * j;
        std::cout << b[j] << " ";
    }

    std::cout << endl << endl;
    std::cout << "The massive a: ";

    for (int i = 0; i < 10; i++) {
        a[i] = 2 * i + 22; // Призначаємо значення у масив a
        std::cout << a[i] << " ";
    }

    std::cout << endl << endl;

    for (int k = 0; k < 10; k++) { // Призначаємо спільні значення масивів a і b у масив c
        for (int f = 0; f < 10; f++) {
            if (a[k] == b[f]) {
                c[k] = int(a[k]);
            }
        }
    }

    std::cout << "The common elements is: ";

    for (int w = 0; w < 10; w++) {
        std::cout << c[w];
    }
}

```

```

for (int w = 0; w < 10; w++) {
    std::cout << c[w];
}

std::cout << endl << endl;

for (int r = 0; r < 10; r++) {
    // Шукаємо максимальний елемент
    if (storageMax < c[r]) {
        storageMax = c[r];
    }
    std::cout << "storageMax = " << storageMax << endl;
}

storageMin = storageMax;
int counter = 9;

do { storageMin = c[counter]; // Шукаємо мінімальний елемент
    counter--;
    std::cout << "storageMin = " << storageMin << endl; }
while (c[counter] > 0);

std::cout << endl << "The sum of minimal and maximal codes = " << storageMax + storageMin;

cin.ignore(1, '\n');
cin.get();
}

```

Тестування програми

```
The massive b: 0 . , * ( & $ " ▲  
The massive a: - ↑ → L ▲ " $ & (  
The common elements is: ▲ "$&(  
  
storageMax = 0  
storageMax = 0  
storageMax = 0  
storageMax = 0  
storageMax = 30  
storageMax = 32  
storageMax = 34  
storageMax = 36  
storageMax = 38  
storageMax = 40  
storageMin = 40  
storageMin = 38  
storageMin = 36  
storageMin = 34  
storageMin = 32  
storageMin = 30  
  
The sum of minimal and maximal codes = 70
```

Висновки: отже, ми навчилися використовувати методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях, та набули навички їх використання. При заданих кодах пошуку елементів, ми знайшли максимальне та мінімальне значення і отримали їх суму - 70.