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 24

Виконав студент ІП-14 Прокопенко Олексій Анатолійович

(шифр, прізвище, ім'я, по батькові)

Перевірив доц. 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

невпорядкованих послідовностях та набути практичних навичок їх використання під час складання програмних специфікацій.

**Варіант 24**

**Побудова математичної моделі:**

Спочатку ініціалізуємо три масиви та заповнимо їх елементами, використовуючи формули з умови (і \* і + 76; 80 - і), змінюючи значення і від

0 до 9. Наступним кроком ми за допомогою вкладеного циклу будемо порівнювати елементи цих масивів і, якщо array1[i] буде дорівнювати array2[j], то цей елемент поміщатимемо в третій масив. Наостанок перевіримо кожен елемент третього масиву на подільність його коду на 2 (ord(array3[i]) % 2 ==

0) та додамо до відповіді.

**Таблиця змінних**

|  |  |  |  |
| --- | --- | --- | --- |
| Опис | Тип | Ім’я | Призначення |
| Масив 1 | chr | array1 | Проміжні дані |
| Масив 2 | chr | array2 | Проміжні дані |
| Масив 3 | chr | array3 | Проміжні дані |
| Лічильник 1 | int | i | Проміжні дані |
| Лічильник 2 | int | j | Проміжні дані |
| Лічильник 3 | int | y | Проміжні дані |
| Сума парних кодів | int | sum | Вихідні дані |

**Розв’язання:**

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

**Псевдокод:**

**підпрограма**

**initialization**(array1, array2):

**повторити** для і від 0 до 10 array1[i] = chr(i \* i + 76) array2[i] = chr(80 - i)

**все повторити все підпрограма**

**підпрограма**

**thirdarray**(array1, array2): y = 0

**повторити** для і від 0 до 10

**повторити** для j від 0 до 10

**якщо** array1[i] == array2[j] array3[y] = array1[i] y = y + 1

**все якщо все повторити**

**все повторити все підпрограма**

**підпрограма**

**countsum**(array3):

sum = 0

**повторити** для і від 0 до 10

**якщо** ord(array3[i]) % 2 == 0

sum = sum + ord(array3[i])

**все якщо все повторити повернути** sum

**все підпрограма**

**підпрограма**

**output**(array):

**повторити** для і від 0 до 10

**виведення** array[i]

**все повторити все підпрограма**

**початок**

кінець

повернути sum

кінець

array1 = [chr(0)] \* 10 array2 = [chr(0)] \* 10 array3 = [chr(0)] \* 10

**initialization**(array1, array2) **thirdarray**(array1, array2) **output**(array1) **output**(array2) **output**(array3)

sum = **countsum**(array3) **виведення** sum

**кінець**

**Блок – схема**

початок

initialization (array1, array2)

i = 0, 10, -

1

+

array1[i] =

chr(i \* i +76)

array2[i] = chr(80 - i)

кінець

thirdarray (array1, array2)

y = 0

i = 0, 10, -

1

+

j = 0, 10, -

1

+

+

array1[i] ==

array2[j]

-

array3[y] = array1[i]

y = y +1

сountsum

(array3)

sum = 0

i = 0, 10, -

1

+

+ ord(array3[i] % -

2 == 0

sum = sum +

ord(array3[i])

output(array)

i = 0, 10, -

1

+

виведення

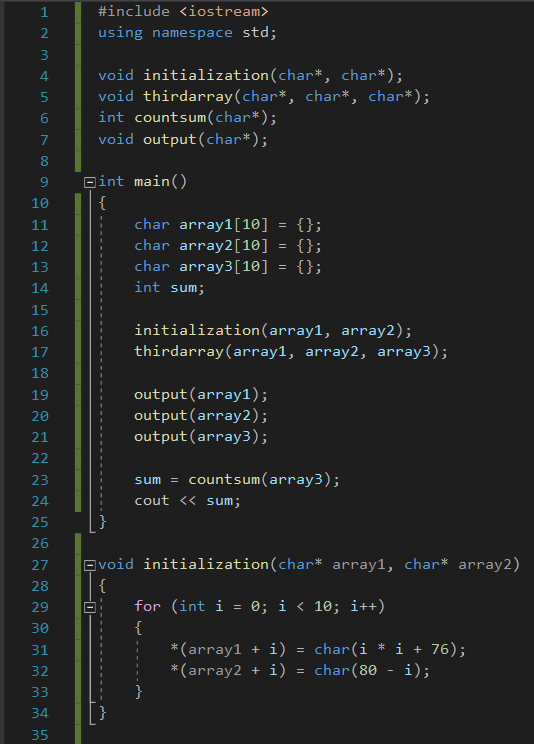
array[i]

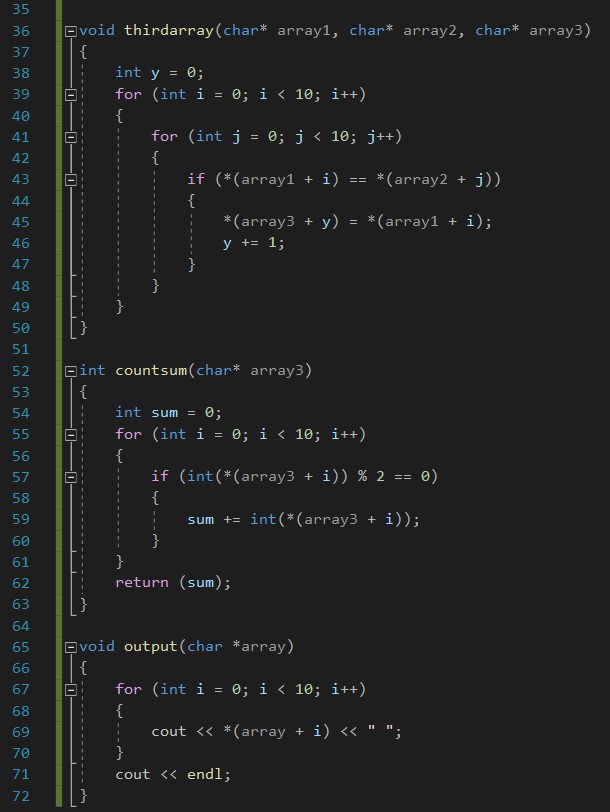
|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| array1 = [chr(0)] \* 10 | | | |
|  | |  | |
| array2 = [chr(0)] \* 10 | | | |
|  | |  | |
| array3 = [chr(0)] \* 10 | | | |
|  | |  | |
|  | initialization (array1, array2) | |  |
|  | |  | |
|  | thirdarray (array1, array2) | |  |
|  | |  | |
|  | output  (array1) | |  |
|  | |  | |
|  | output (array2) | |  |
|  | |  | |
|  | output  (array3) | |  |
|  | |  | |
| sum =  countsum(array3) | | | |
|  | |  | |

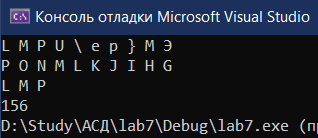
виведення

sum

кінець

**Код програми**



**Тестування**

**Висновки:**

Я набув практичних навичок використання алгоритмів лінійного пошуку в послідовностях під час складання програмних специфікацій. В процесі виконання лаболаторної роботи я навчився використовувати масиви, описувати її у псевдокоді та позначати на блок-схемі.