

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 20

Виконав студент

ІП-15, Ликова Катерина Олександрівна
(шифр, прізвище, ім'я, по батькові)

Перевірив

(прізвище, ім'я, по батькові)

Київ 2021

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

Постановка задачі

Описати три змінні індексованого типу з 10 символьних значень. Ініціювати дві змінні виразами: $(130 - i)$ та $(120 + i)$. Ініціювати третю змінну рівними значеннями двох попередніх змінних. Серед значень третьої змінної знайти кількість елементів, коди яких менше 127.

Математична модель

Змінна	Тип	Ім'я	Призначення
Перша змінна	Символьний	a[]	Проміжні дані
Друга змінна	Символьний	b[]	Проміжні дані
Третя змінна	Символьний	c[]	Проміжні дані
Параметр циклів	Цілий	i	Проміжні дані
Кількість елементів, коди яких менше 127	Цілий	k	Результат

Для того, щоб знайти серед значень третьої змінної кількість елементів, коди яких менше 127, потрібно присвоїти змінній k значення 0, а змінній c - значення десяти нулів. Потім треба знайти значення змінних a та b за формулами $(130 - i)$ та $(120 + i)$, знайти рівні значення a та b, присвоїти ці значення змінній c. Далі необхідно серед отриманих значень у змінній c відібрати ті, що не дорівнюють 0 та менші за 127. Після кожної ітерації, під час якої умови задовольняються, змінна k збільшується на 1, таким чином отримую потрібний результат.

Розв'язання

Крок 1. Визначимо основні дії.

Крок 2. Присвоюємо значення змінній k.

Крок 3. Присвоюємо значення змінній c[].

Крок 4. Деталізація дії знаходження a[].

Крок 5. Деталізація дії знаходження b[].

Крок 6. Деталізація дії знаходження c[].

Крок 7. Деталізація дії знаходження k.

Псевдокод

крок 1

початок

присвоєння значення змінній k

присвоєння значення змінній c[]

деталізація дії знаходження a[]

деталізація дії знаходження b[]

деталізація дії знаходження c[]

деталізація дії знаходження k

кінець

крок 2

початок

k=0

присвоєння значення змінній c[]

деталізація дії знаходження a[]

деталізація дії знаходження b[]

деталізація дії знаходження c[]

деталізація дії знаходження k

кінець

крок 3

початок

k=0

повторити для i від 0 до 9

c[i]=0

все повторити

деталізація дії знаходження a[]

деталізація дії знаходження b[]

деталізація дії знаходження $c[]$

деталізація дії знаходження k

кінець

крок 4

початок

$k=0$

повторити для i від 0 до 9

$c[i]=0$

все повторити

повторити для i від 0 до 9

$a[i] = 130 - i$

деталізація дії знаходження $b[]$

деталізація дії знаходження $c[]$

все повторити

деталізація дії знаходження k

кінець

крок 5

початок

$k=0$

повторити для i від 0 до 9

$c[i]=0$

все повторити

повторити для i від 0 до 9

$a[i] = 130 - i$

$b[i] = 120 + i$

деталізація дії знаходження $c[]$

все повторити

деталізація дії знаходження k

кінець

крок 6

початок

$k=0$

повторити для i від 0 до 9

$c[i]=0$

все повторити

повторити для i від 0 до 9

$a[i] = 130 - i$

$b[i] = 120 + i$

якщо $a[i] == b[i]$

то

$c[i] = a[i]$

все якщо

все повторити

деталізація дії знаходження k

кінець

крок 7

початок

$k=0$

повторити для i від 0 до 9

$c[i]=0$

все повторити

повторити для i від 0 до 9

$a[i] = 130 - i$

$b[i] = 120 + i$

якщо $a[i] == b[i]$

то

$c[i] = a[i]$

все якщо

все повторити

повторити для i від 0 до 9

якщо $c[i] < 127$

то

якщо $!c[i] == 0$

то

$k=k+1$

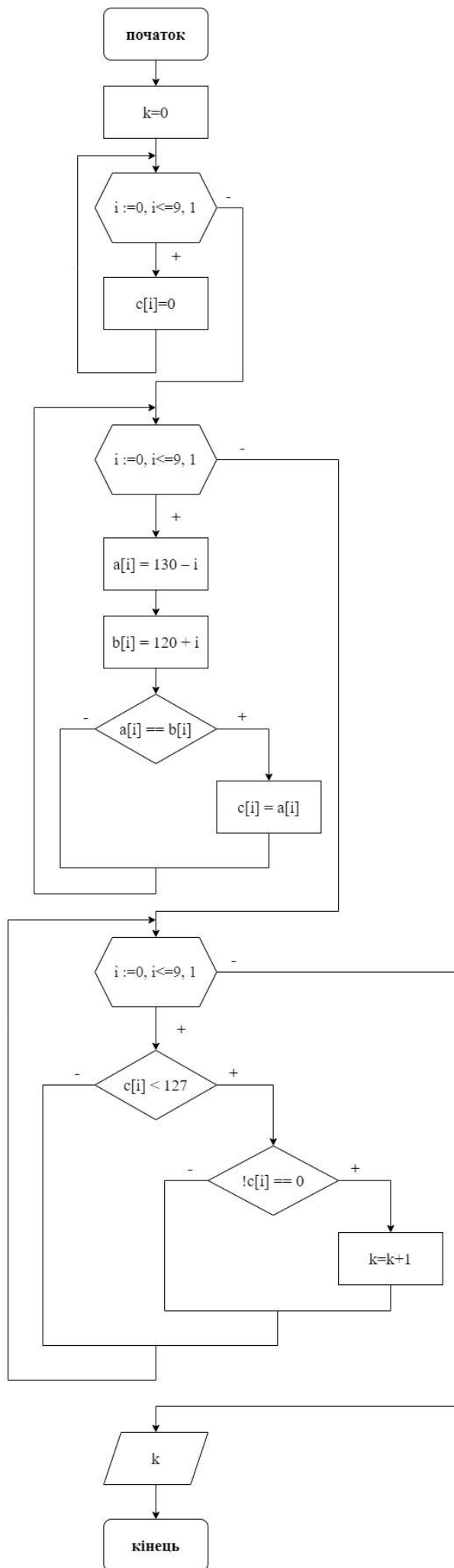
все якщо

все якщо

все повторити

кінець

Блок-схема



Код

```
1  #include <iostream>
2  #include <iomanip>
3  int main()
4  {
5      char a[10], b[10], c[10];
6      int k = 0;
7      for (int i = 0; i < 10; i++)
8      {
9          c[i] = 0;
10     }
11     for (int i = 0; i < 10; i++)
12     {
13         a[i] = 130 - i;
14         b[i] = 120 + i;
15         if (a[i] == b[i])
16         {
17             c[i] = a[i];
18         }
19     }
20
21     for (int i = 0; i < 10; i++)
22     {
23         if (c[i] < 127)
24         {
25             if (!c[i] == 0)
26             {
27                 k++;
28             }
29         }
30     }
31
32     std::cout << "k = " << k << std::endl;
33     system("pause");
34 }
```

Тестування

```
k = 1
Для продолжения нажмите любую клавишу . . .
```


Висновки: Для побудови алгоритму розв'язання заданої задачі я застосувала алгоритм лінійного пошуку для створення програмної специфікації. Завдяки цьому я вивчила властивості формул пошуку даних та навчилась створювати символічні послідовності, обробляти їх згідно з умовою та використовувати їх на практиці.