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 арифметичних циклічних алгоритмів» Варіант 17

Виконав студент <u>ІП-15 Куркчі Юрій Сергійович</u>

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

Варіант 17

№	Вираз для обчислення елемента		Знайти	
	1-го масиву	2-го масиву		
17	5 * i + 25	55 - 5 * i	Добуток елементів, коди яких менше 82	

Постановка задачі

Заданий алгоритм повинен:

- 1. Створити 2 змінні індексованого типу з 10 символьних значеннь.
- 2. Ініціювати перші 2 змінні за формулами 5i + 25 та 55 5i, де $i \in [1,10]$.
- 3. Створення третьої змінної та ініціювати її спільними значеннями двох попередніх.
- 4. Знайти добуток елементів з третьої змінної, коди яких менше 82.

Побудова математичної моделі

Таблиця змінних

Змінна	Тип	Ім'я	Призначення
Розмірність масивів	Натуральний	Z	Початкові дані
Перший масив	Символьний	F	Проміжні дані
Другий масив	Символьний	S	Проміжні дані
Третій масив	Символьний	Т	Проміжні дані
Додаткові масиви	Символьний	masiv1,masiv2,masiv3	Проміжні дані

Добуток кодів	Натуральний	mul	Проміжні дані
Результат добутку	Цілий	result	Кінцеві дані
Символьний	Логічний	charesult	Кінцеві дані
результат добутку			

Власні функції

1 f1(одновимірний символьний масив) — створює й виводить перший символьний масив 2 s1(одновимірний символьний масив) — створює й виводить другий символьний масив 3 t3(одновимірний символьний масив, одновимірний символьний масив, одновимірний символьний масив, одновимірний символьний масив) — створює й виводить третий символьний масив згенерований на основі перших двух. 4 min82 (одновимірний символьний масив, одновимірний символьний масив, одновимірний символьний масив) — повертає добуток елементів, коди яких менше 82.

- 1. Згенеруємо перші 2 масиви за формулами 5i + 25 та 55 5i.
- 2. Згенеруємо третій масив шляхом перевірки кожної пари елементів на рівність та їх внесення до масиву. Інші елементи заповнимо нулями.
- 3. Розрахуємо добуток ненульових елементів третього масиву.

Розв'язання

- 1. Визначимо основні дії.
- 2. Деталізуємо дію визначення та виведення першого масиву.
- 3. Деталізуємо дію визначення та виведення другого масиву.

4.

5. Деталізуємо дію визначення та виведення третього масиву.

6. Деталізуємо дію обрахунку та виведення добутків ненульових елементів третього масиву.

Псевдокод алгоритму

```
Крок 9
```

```
Початок
    Z := 10
f1(F)
s2(S)
t2(F,S,T)
min82(T)
Кінець
Підпрограма
    f1(masiv1)
      повторити
     для і від 1 до Z
     masiv1[i]:=char(25+5*i)
     F[i] = masiv1[i]
      вивід S[i]
      все повторити
Все підпрограма
Підпрограма
    s2(masiv2)
      повторити
      для і від 1 до Z
      masiv2[i]:=char(55-5*i)
      S[i] = masiv2[i]
```

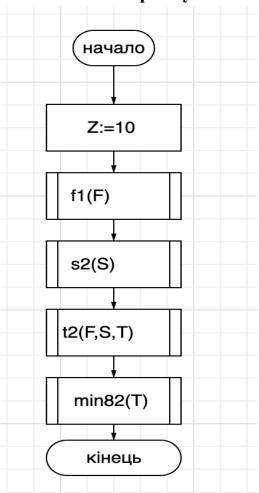
вивід S[i]

Все підпрограма

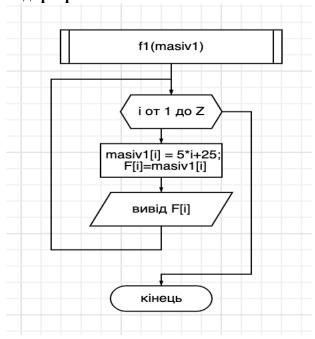
все повторити

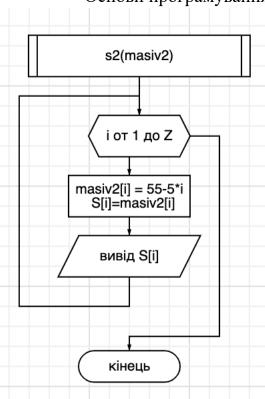
```
Підпрограма
t3(masiv1[],masiv2[],masiv3[])
   повторити
    для і від 1 до Z
     повторити
      для ј від 1 до
          якщо masiv1[i]==masiv2[j]
           T0
           masiv3[i]=masiv1[i]
           T[i]=masiv3[i]
           Виведення Т[і]
           все якщо
           все повторити
Підпрограма
    Min82(masiv3[])
mul:=1
   повторити
   для і від 1 до Z
     якщо
       masiv3[i]!=0
       &&
       masiv[i]!=0
       mul=mul*masiv3[i]
       result=mul
      все якщо
    все повторити
 charesult=result%127
 вивести result
 вивксти charesult
Все підпрограма
```

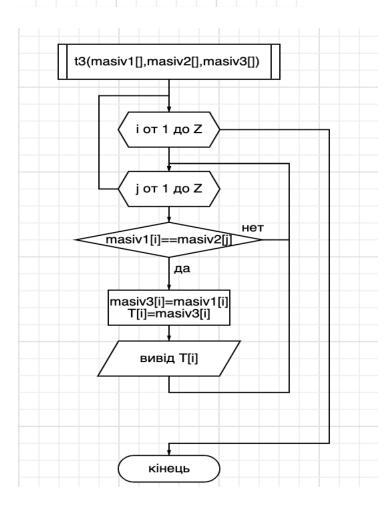
Блок-схема алгоритму

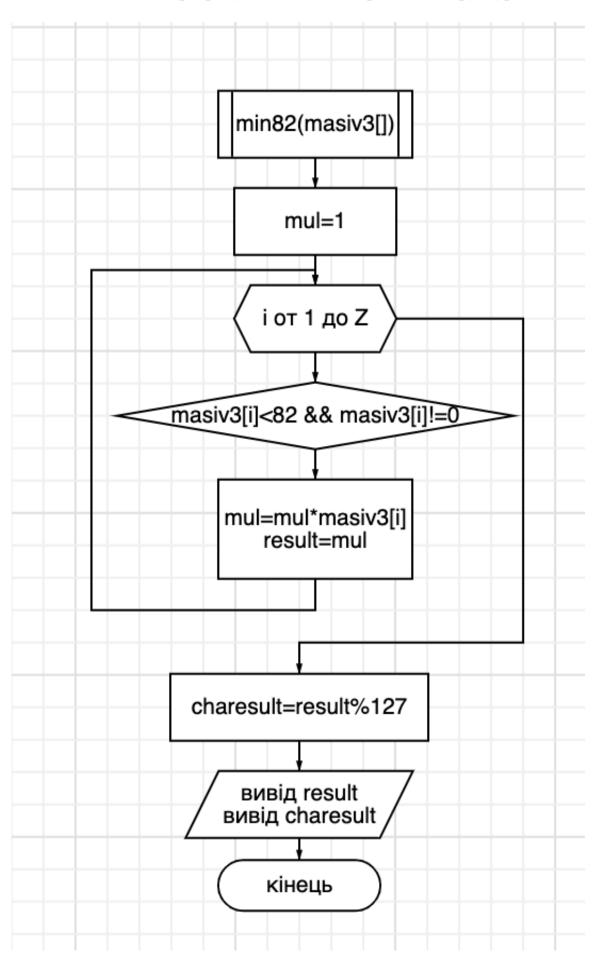


Підпрограми









Код програми

```
#include <iomanip>
#include <ctime>
#include <iostream>
#include <math.h>
using namespace std;
const int Z = 10;
char F[Z], S[Z], T[Z];
  void f1(char[]);//введення масивів
  void s2(char[]);//вивід
  void t3(char[], char[]);
  int min82(char[]);
  int main(){
cout << "Масиви:" << "\n\n"<< "First: ";
f1(F);
cout << "\n\n\n" << "Second: ";</pre>
s2(S);
cout << "\n\n\n" <<" Thitd: ";</pre>
t3(F,S,T);
cout << "\n\n\n";</pre>
min82(T);
cout << "\n\n\n";</pre>
return 0;
```

```
void f1(char masiv1[]){
\vee for(int i = 1; i \ll Z; i++){
    masiv1[i] = 5*i+25;
   F[i]=masiv1[i];
   cout << F[i] << ", ";
   }
  }
void s2(char masiv2[]){
\vee for(int i = 1; i \ll Z; i++){
   masiv2[i] = 55-5*i;
   S[i]=masiv2[i];
   cout << S[i] << ", ";
   }

∨ void t3(char masiv1[], char masiv2[], char masiv3[]){
  for(int i = 1; i \le Z; i++)
    for(int j=1; j \ll Z; j++)
        if(masiv1[i]==masiv2[j]){
          masiv3[i]=masiv1[i];
          T[i]=masiv3[i];
          cout << T[i] << ", ";
```

```
int min82(char masiv3[]){
  int mul=1;
 int result;
  cout << "Елементи третього масиву: ";
   for(int i = 1; i \ll Z; i++){
     int q=masiv3[i];
     cout << q << " ";
   if(masiv3[i]<82 && masiv3[i]!=0)
      mul =mul * masiv3[i];
      result=mul;
   }
char charesult=result%127;
cout << "\n\n" << "результат добутку елементів: " << result;
cout << "\n\n" << "символьний результат добутку елементів: " << charesult;
return result;
}
```

```
Масиви:

First: , #, (, -, 2, 7, <, A, F, K,

Second: 2, -, (, #, , , ,

, , ,

Third: , #, (, -, 2,

Елементи третього масиву: 30 35 40 45 50 0 0 0 0 0

результат добутку елементів: 94500000

символьний результат добутку елементів: >
```

Випробування алгоритму

Блок	Дія
	Початок
1.	size:=10
2.	f1 (F)
3.	i=1, i <= 10, i++
4.	masiv1:=char(30)='RS'
5.	i=2, i <= 10, i++
6.	masiv1:=char(35)='#'
7.	i=3, i <= 10, i++
8.	masiv1:=char(40)='('
9.	i=4, i <= 10, i++
10.	masiv1:=char(45)='-'
11.	i=5, i <= 10, i++
12.	masiv1:=char(50)='2'
13.	i=6, i <= 10, i++
14.	masiv1:=char(55)='7'
15.	i=7, i <= 10, i++
16.	masiv1:=char(60)='<'
17.	i=8, i <= 10, i++
18.	masiv1:=char(65)='A'
19.	i=9, i <= 10, i++
20.	masiv1:=char(70)='F'
21.	i=10, i <=10, i++
22.	masiv1:=char(75)='K'
23.	i=11,i>=10
24.	$F:=\{RS,\#,(,-,2,7,<,A,F,K\}$
25.	s2(S)
26.	i=1,i<=10,i++
27.	masiv2:=char(50)='2'
28.	i=2,i<=10,i++

	Ochobit hporpamybaimit 1.7 km ophitimi ta orpyktyph
29.	masiv2:=char(45)='-'
30.	i=3, i <= 10, i++
31.	masiv2:=char(30)='('
32.	i=4, i <= 10, i++
33.	masiv2:=char(35)='#'
34.	i=5,i<=10,i++
35.	masiv2:=char(30)='RS'
36.	i=6, i <= 10, i++
37.	masiv2:=char(25)='EM'
38.	i=7,i<=10,i++
39.	masiv2:=char(20)='NAK'
40.	i=8,i<=10,i++
41.	masiv2:=char(15)='SI'

42.	i=9,i<=10,i++
43.	masiv2:=char(10)='LF'
44.	i=10,i<=10,i++
45.	masiv2:=char(5)='ENQ'
46.	i=11,i>=10
47.	$S:=\{2,-,(,\#,RS,EM,NAK,SI,LF,ENQ)\}$
48.	t3(T)
49.	thirdArray={RS,#,(,-,2,NUL, NUL, NUL, NUL, NUL}
50.	Min82
51.	mul:=1
52.	i=1,i<=10,i++
53.	'RS'!='NUL'
54.	mul*=30=30
55.	i=2,i<=10,i++
56.	'#'!='NUL'
57.	mul *=35=1050
58.	i=3,i<=10,i++
59.	'('!='NUL'
60.	mul *=40=4200
61.	i=4,i<=10,i++
62.	'-'!='NUL'
63.	mul *=45=1890000
64.	i=5, i<=10, i++

	Сопори програмувания 1. Азигоритми та отруктури	
65.	'2'!='NUL'	
66.	mul *=50=94500000	
67.	i=6, i <= 10, i++	
68.	'NUL' =='NUL'	
69.	i=7,i<=10,i++	
70.	'NUL' =='NUL'	
71.	i=8, i<=10, i++	
72.	'NUL' =='NUL'	
73.	i=9,i<=10,i++	
74.	'NUL'=='NUL'	
75.	i=10, i <=10, i++	
76.	'NUL'=='NUL'	
77.	i=11,i>10	
78.	result=94500000	
79.	Churesult=resul%127	
80.	churesult:=char(62):= '>'.	
81.	Виведення 94500000	
	Виведення '>'	
82.		
	Кінець	

Висновки

Протягом виконання цієї лабораторної роботи я набув навичок використання методів послідовного пошуку у невпорядкованих послідовностях та практичних навичок їх використання під час складання програмних специфікацій. Маючи формули задання елементів двох одновимірних символьних масивів, я склав програму яка успішно знаходить масив, який складається з спільних елементів попередніх масивів та добуток його ненульових елементів.