

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 структури даних»
«Дослідження лінійного пошуку в послідовностях»
Варіант 34

Виконав студент ІІ-1134 Шамков Іван Дмитрович
(прізвище, ім'я, по батькові)

Перевірив викладач Мартинова Оксана Петрівна
(прізвище, ім'я, по батькові)

Київ 2021
Лабораторна робота №7
Дослідження лінійного пошуку в послідовностях

Лабораторна робота 7

Дослідження лінійного пошуку в послідовностях

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

Варіант: 34

Умова задачі:

Завдання

Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опису трьох змінних індексованого типу з 10 символних значень.
2. Ініціювання двох змінних виразами згідно з варіантом (табл. 1).
3. Ініціювання третьої змінної рівними значеннями двох попередніх змінних.
4. Обробки третьої змінної згідно з варіантом.

34	$44 + 2 * i$	$55 - 2 * i$	Кількість елементів, коди яких діляться на 3
----	--------------	--------------	--

Математична модель:

Змінна	Тип	Ім'я	Призначення
Масив 1	Символьний	A	Початкове дане
Масив 2	Символьний	B	Проміжне значення
Масив 3	Символьний	C	Проміжне значення
Лічильник	Цілий	i	Проміжне значення
Лічильник	Символьний	j	Проміжне значення
Значення для підрахунку членів масиву	Цілий	b	Проміжне значення
Значення для підрахунку членів масиву	Цілий	k	Проміжне значення
Початок, з якого починаємо знаходити елементи масиву	Цілий	start	Проміжне значення

Значення для підрахунку кількості елементів, коди яких націло діляться на три	Цілий	count	Результат
---	-------	-------	-----------

Постановка задачі:

Отже, математичне формулювання нашої задачі полягає в тому, щоб створити два масиви розміром 10 за формулою $b+k*i$, де i – лічильник. Значення b та k залежать від масиву. Для A $b=44$ $k=2$, а для B $b=55$ $k=(-1)$. Трохи змінимо умову для масиву B , щоб мати хоч якісь спільні елементи між A та B . Після цього створюємо третій масив C , заповнюючи його спільними елементами масивів A та B . Усе це робимо через лінійний пошук. Потім пробігаємося по третьому масиву та шукаємо елементи, коди яких діляться націло на три. Усе це виводимо.

Вважатимемо $\text{ord}(x)$ отримання коду символу x у відповідності до таблиці ASCII, а $\text{chr}(x)$, навпаки, переведення числа в символ у відповідності з кодом(числом x) за таблицею ASCII.

$A.\text{append}(x)$ – заповнення масиву A одним елементом x з кінця

Наступні функції є створеними власноруч:

$\text{CreateArray}(b, k, \text{start})$

$\text{ArraySame}(A, B)$

$\text{FindDiv}(C)$

Псевдокод:

Крок 1. Визначимо основні дії.

Крок 2. Деталізуємо значення A , B , C

Крок 3. Пошук елементів, коди яких діляться націло на три

Крок 4. Виведення A , B , C , count

Крок 1:

Start

Деталізуємо значення A, B, C

Пошук елементів, коди яких діляться націло на три

Виведення A, B, C, count

End

Крок 2:

Start

A=CreateArray(44, 2, 5)

B=CreateArray(55, -1, -5)

C=ArraySame(A, B)

Деталізуємо переведення числа у десяткову систему числення

Виведення A, B, C, count

End

Крок 3:

Start

A=CreateArray(44, 2, 5)

B=CreateArray(55, -1, -5)

C=ArraySame(A, B)

count=FindDiv(C)

Виведення A, B, C, count

End

Крок 4:

Start

A=CreateArray(44, 2, 5)

B=CreateArray(55, -1, -5)

C=ArraySame(A, B)

count=FindDiv(C)

CoutArray(A)

CoutArray(B)

CoutArray(C)

output count

End

Підпрограми

CreateArray(b, k, start)

for i from start to start+9

 A.append(chr(b+k*i))

end for

return A

ArraySame(A, B)

for i from 0 to len(A)-1

 for j from 0 to len(B)-1

 if A[i]==B[j]

 C.append(A[i])

 end if

 end for

end for

while len(C)<10:

 repeat

```
        C.append(chr(0))
    end while
    return C
```

FindDiv(C)

```
    for i from 0 to (len(C)-1)
        if ord(C[i])%3==0 and ord(C[i])!=0
            then
                count+=1
            end if
    end for
    return count
```

CoutArray(A)

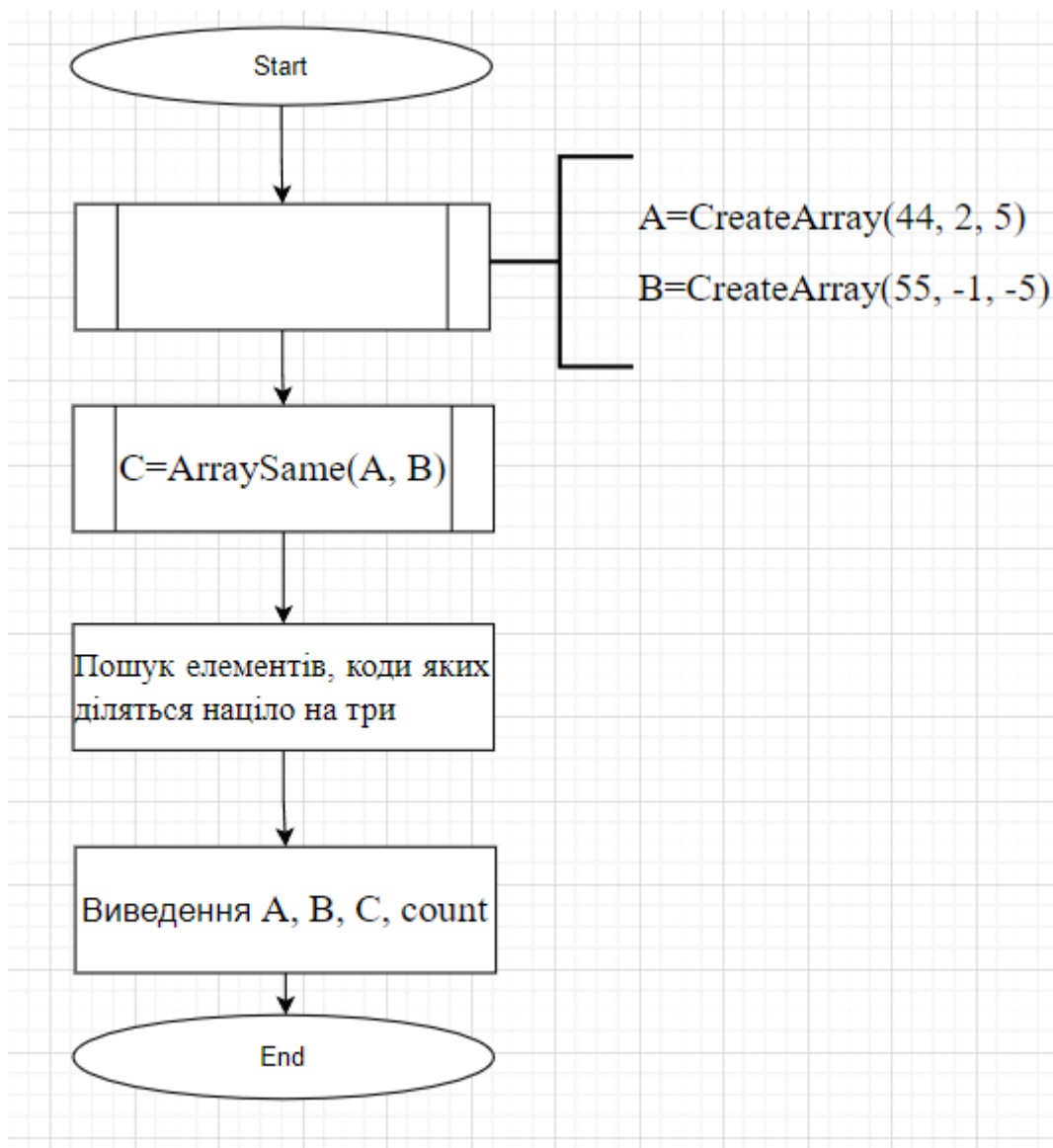
```
    for i from 0 to 9
        repeat
            output A[i]
        end for
    end for
    return
```

Блок схема:

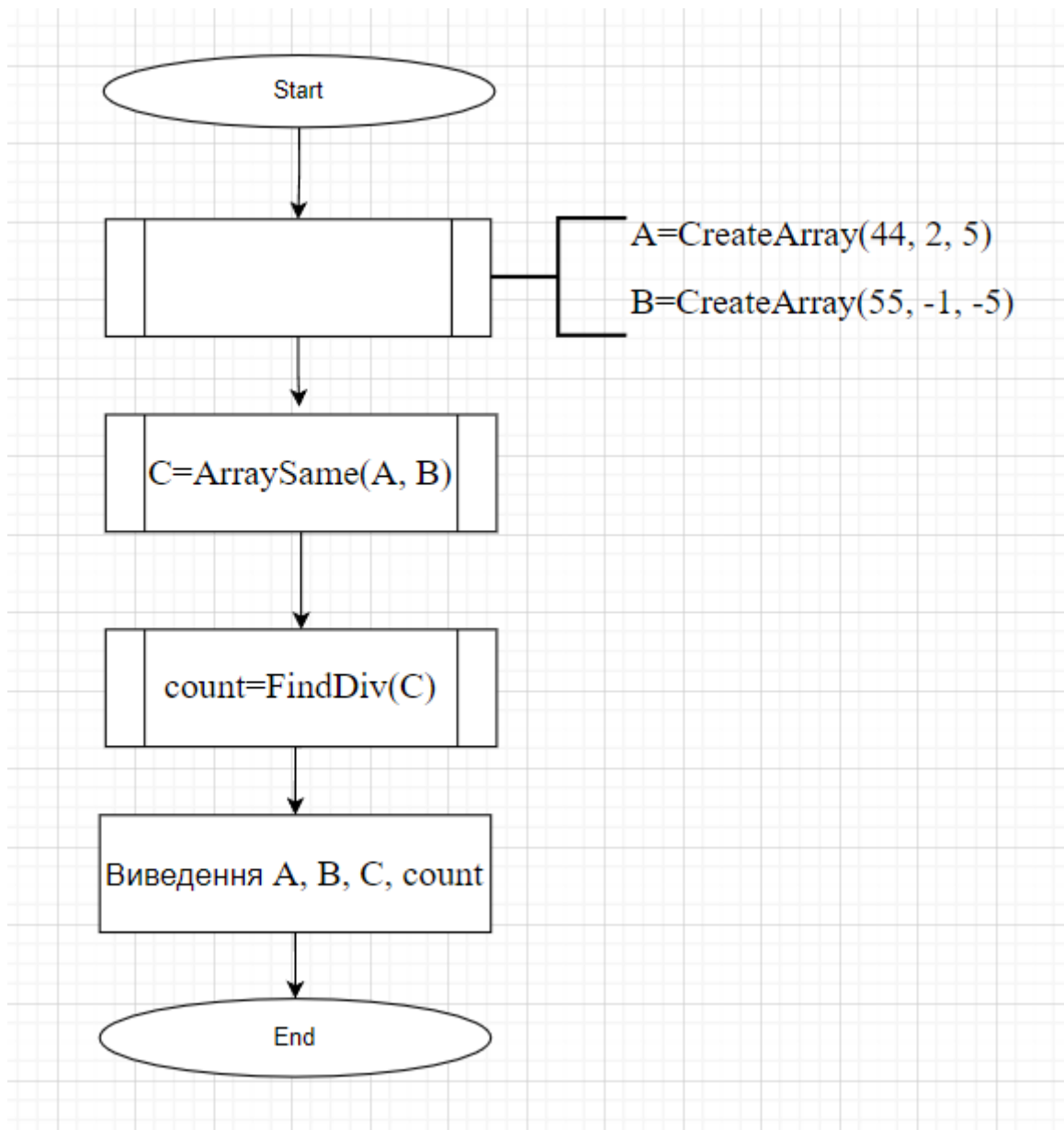
Крок 1



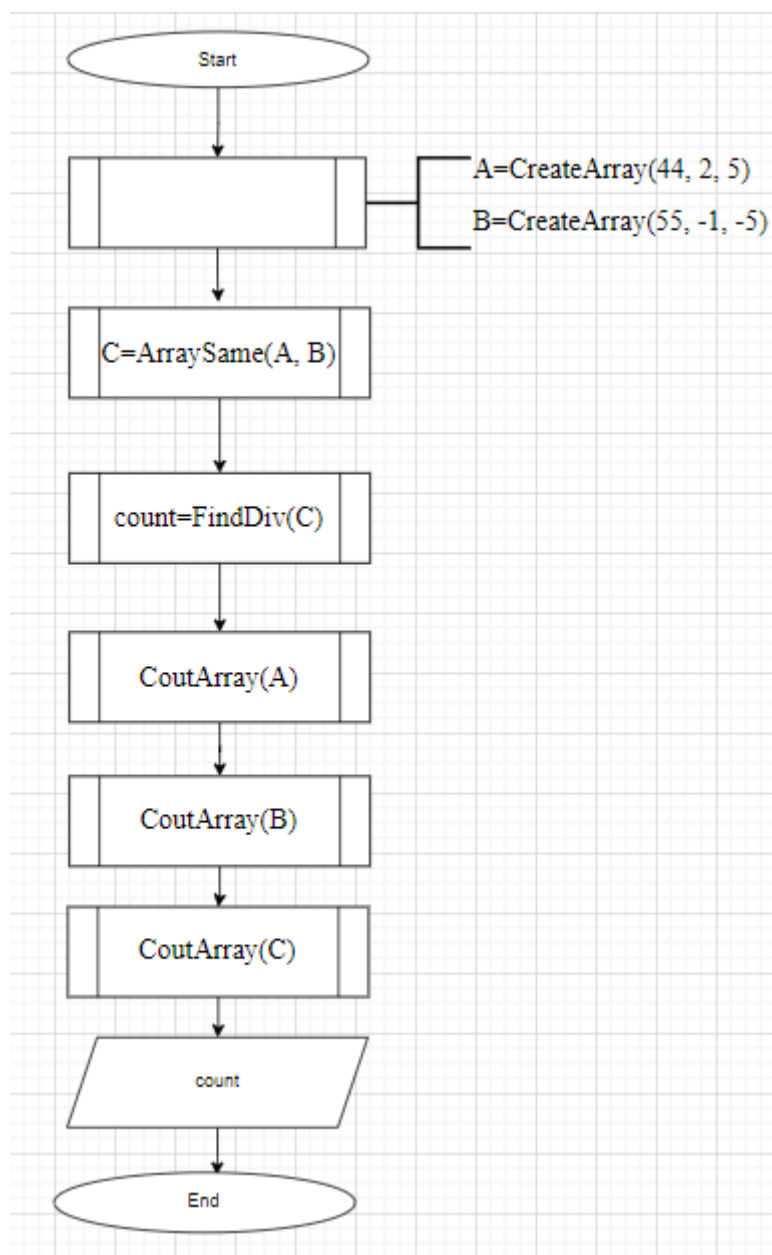
Крок 2



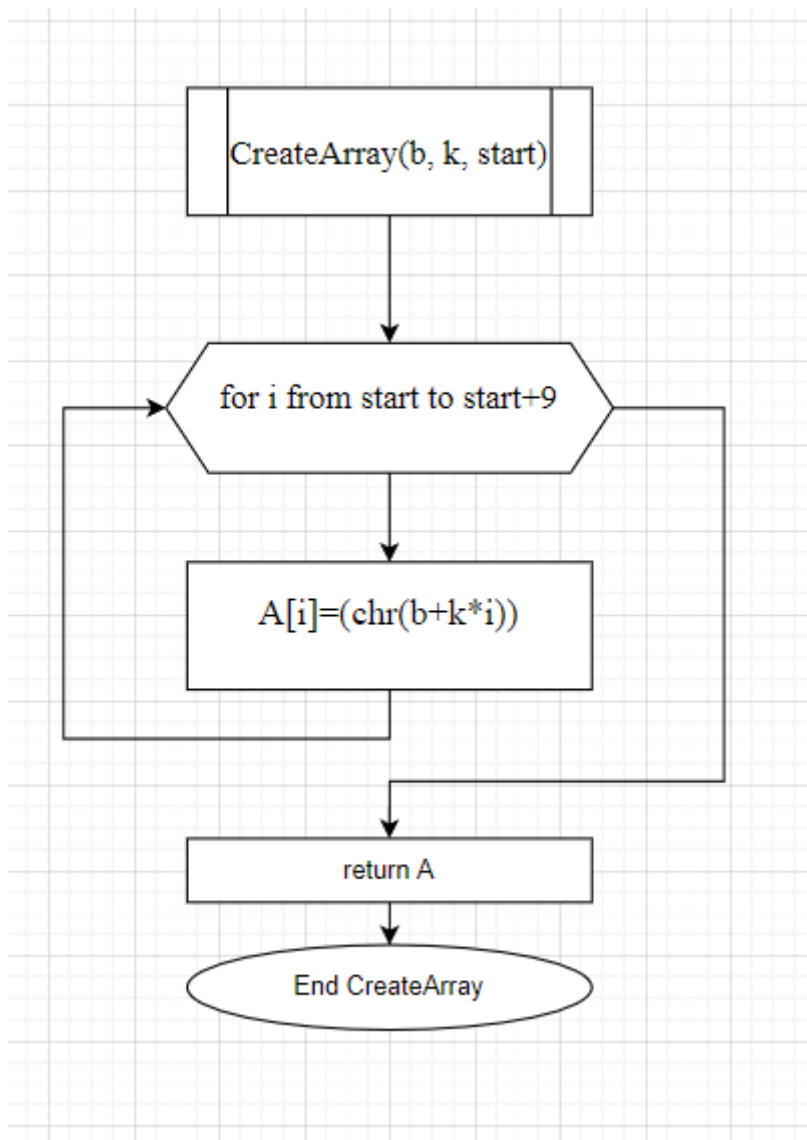
Крок 3

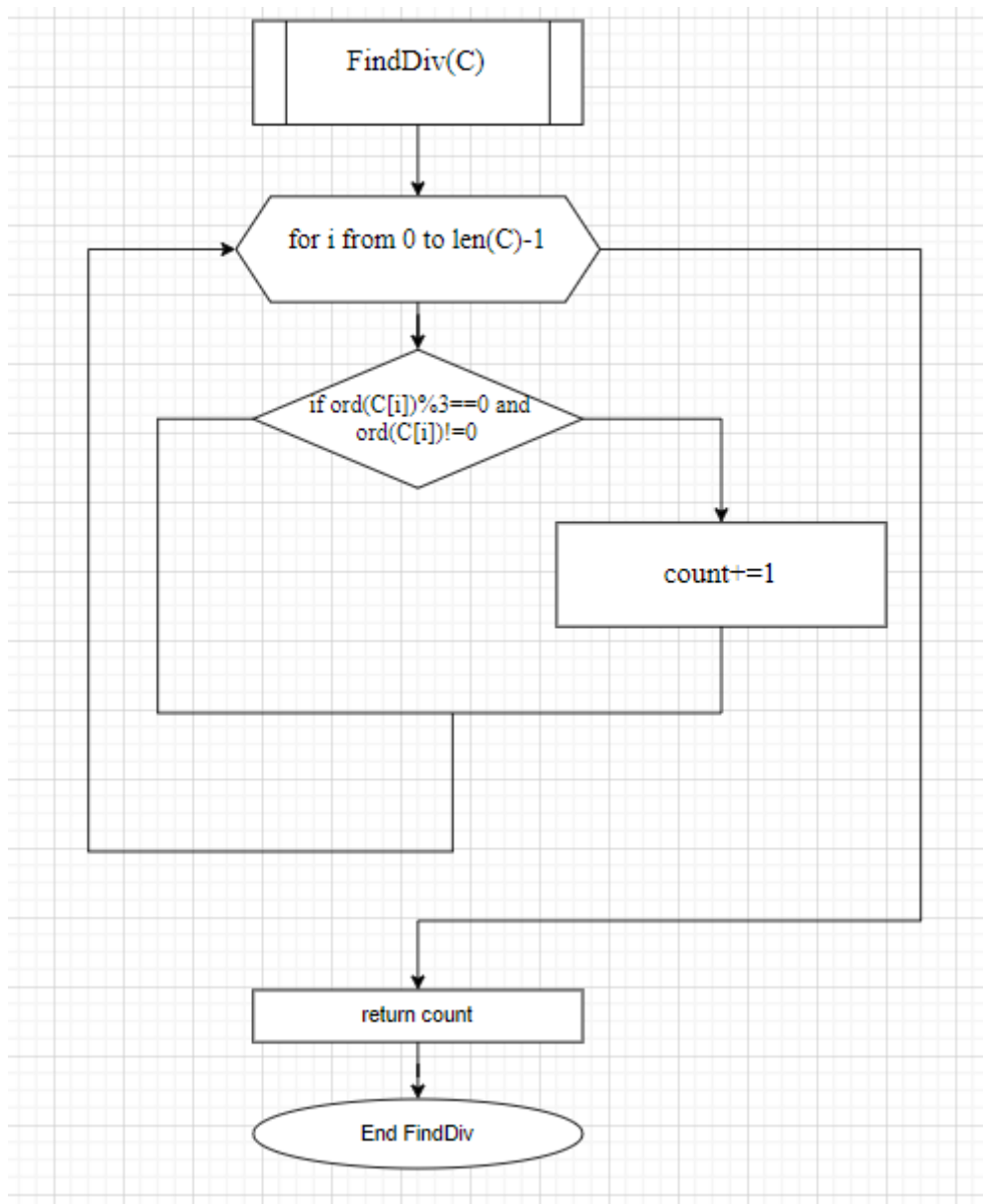


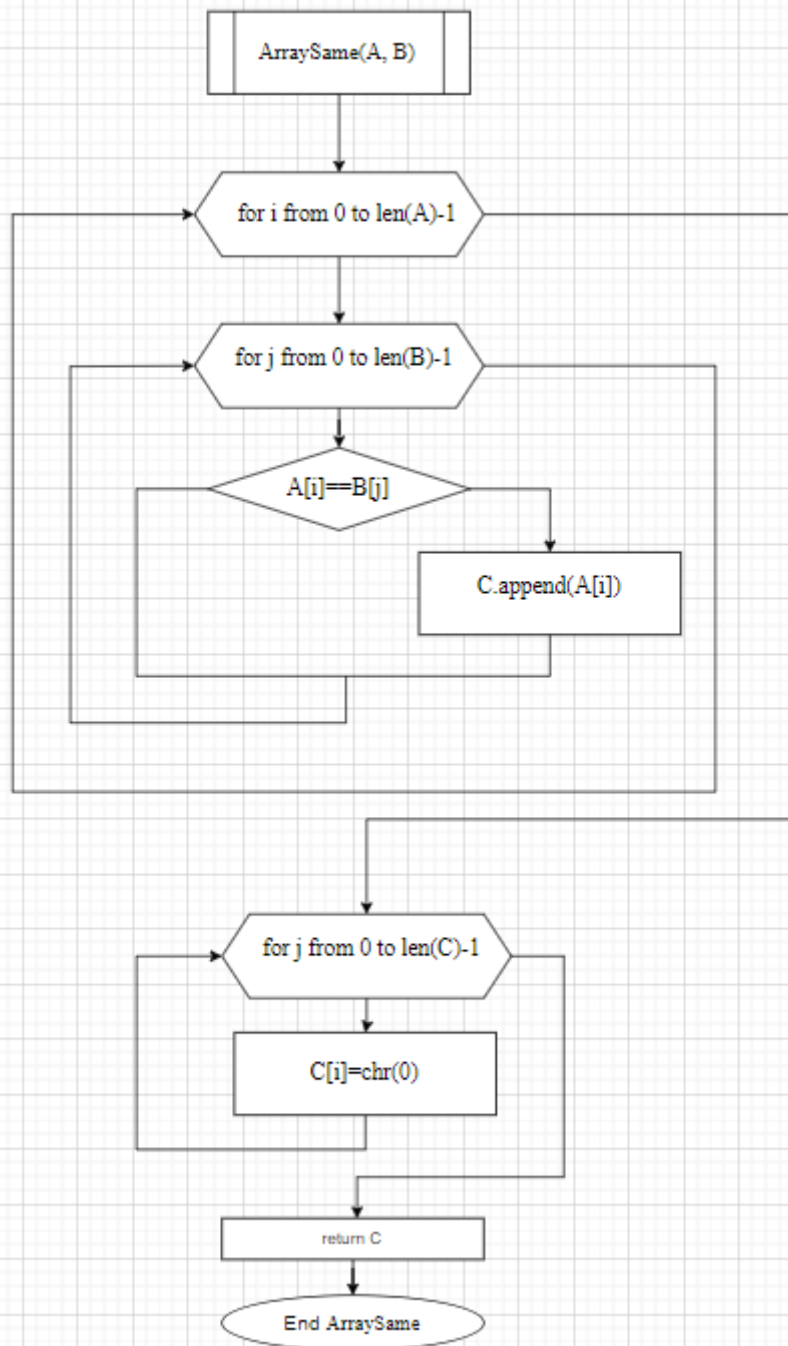
Крок 4

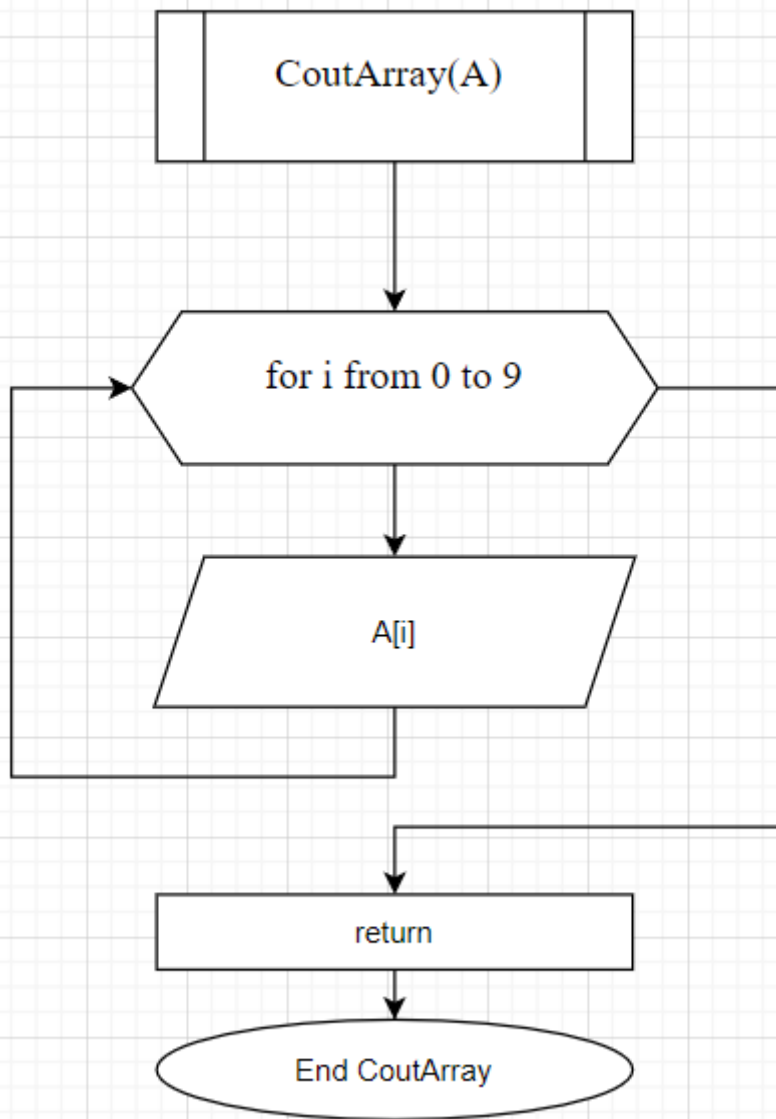


Підпрограми









Код на Python:

```

import string

def CreateArray(b:int, k:int, start:int):
    A=[]
    for i in range(start, start+10):
        A.append(chr(b+k*i))
    return A

```

```

def ArraySame(A, B):

```

```

    C=[]
    for i in A:
        for j in B:
            if i==j:
                C.append(i)
    while len(C)<10:
        C.append(chr(0))
    return C

```

```

def FindDiv(C):

```

```

    count=0
    for i in range (len(C)):
        if ord(C[i])%3==0 and ord(C[i])!=0:
            count+=1
    return count

```

```

A=CreateArray(44, 2, 5)

```

```
B=CreateArray(55, -1, -5)  
C=ArraySame(A, B)  
print(*A, sep=' ')  
print(*B, sep=' ')  
print(*C, sep=' ')  
count=FindDiv(C)  
print("The quantity of such numbers is", count)
```

Копії екранних форм:


```

import string
def CreateArray(b:int, k:int, start:int):
    A=[]
    for i in range(start, start+10):
        A.append(chr(b+k*i))
    return A

def ArraySame(A, B):
    C=[]
    for i in A:
        for j in B:
            if i==j:
                C.append(i)
    while len(C)<10:
        C.append(chr(0))
    return C

def FindDiv(C):
    count=0
    for i in range (len(C)):
        if ord(C[i])%3==0 and ord(C[i])!=0:
            count+=1
    return count

A=CreateArray(44, 2, 5)
B=CreateArray(55, -1, -5)
C=ArraySame(A, B)
print(*A, sep=' ')
print(*B, sep=' ')
print(*C, sep=' ')
count=FindDiv(C)
print("The quantity of such numbers is", count)

```

```

6 8 : < > @ B D F H
< ; : 9 8 7 6 5 4 3
6 8 : <
The quantity of such numbers is 2

```

Випробування алгоритму

Блок	Дія
	Початок
1	A=CreateArray(44, 2, 5) [54, 56, 58, 60, 62, 64, 66, 68, 70, 72]
2	B=CreateArray(55, -1, -5) [60, 59, 58, 57, 56, 55, 54, 53, 52, 51]
3	C=ArraySame(A, B) [54, 56, 58, 60, 0, 0, 0, 0, 0, 0]
4	count=FindDiv(C) count=2(не враховуємо нуль)
	Кінець

Висновок

Отже, виконавши цю лабораторну роботу, ми навчилися використовувати лінійний пошук в масивах. Його реалізація полягає в тому, щоб порівняти кожен елемент масива А з кожним елементом масива В. Це відбувається за допомогою двох арифметичних циклів, один з яких вкладений в інший. У процесі виконання ми сформулювали задачу, побудували математичну модель та псевдокод алгоритму, що допомогло нам краще її зрозуміти. Основною частиною алгоритму є створення трьох масивів. Для перших двох було створено підпрограму, яку відповідно використовуємо двічі. Для третього ж масива використовуємо іншу підпрограму, адже спосіб його задання інший. Після створення трьох масивів, шукаємо в третьому такі коди символів значення, що націло ділять на три. Підраховуємо їхню кількість та в кінці виводимо це користувачу.