

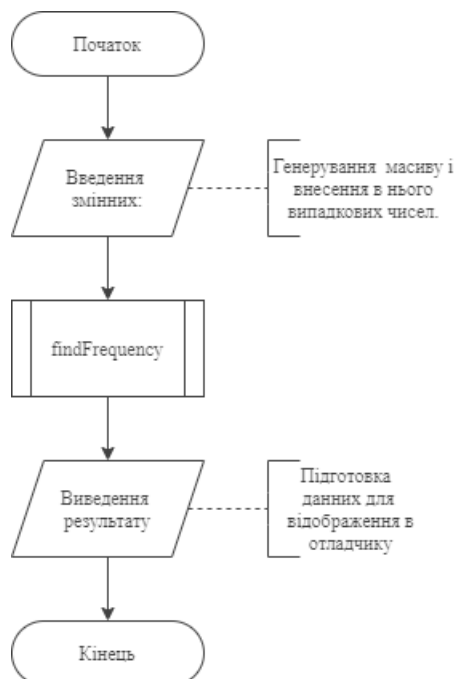
Лабораторна робота №11. Вступ до показчиків.

Вимоги:

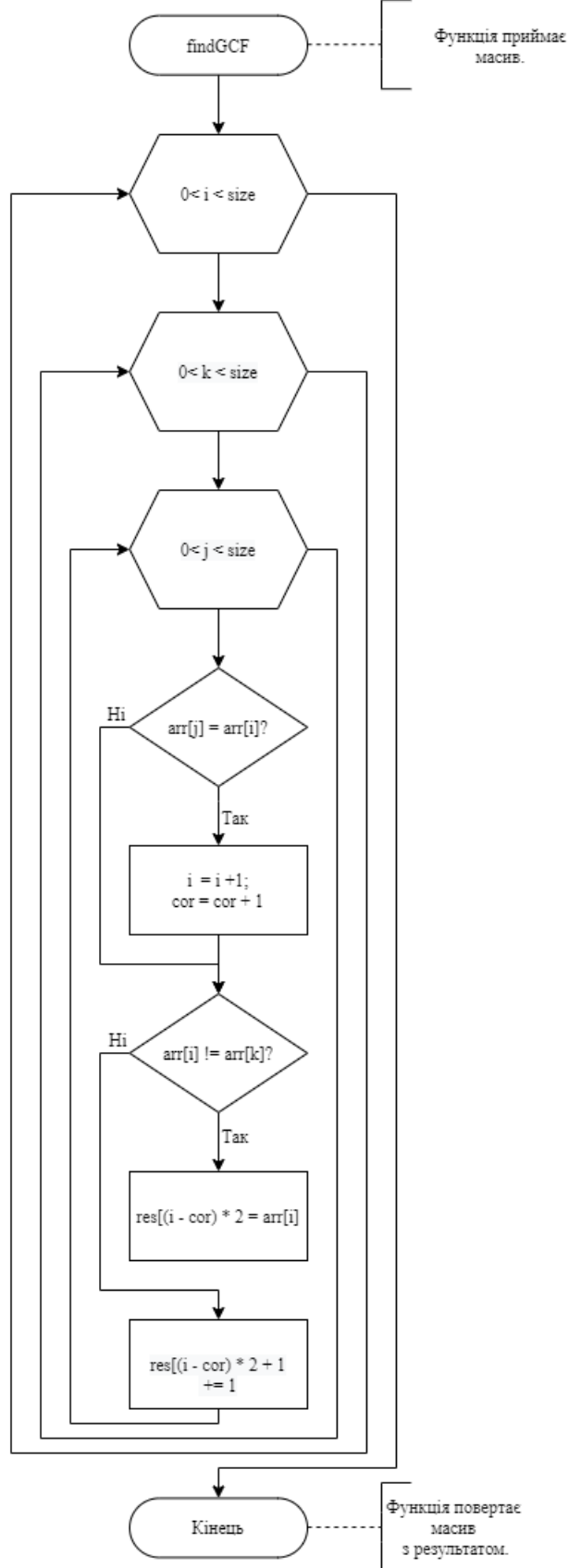
- *Розробник:* Зозуля Ігор студент групи КІТ-120а.
- *Загальне завдання:* Реалізувати програми з використанням функцій.
- *Індивідуальне завдання :* Визначити частоту зустрічання певного елементу в масиві.

Опис програми:

- *Функціональне призначення :* Визначення найбільшого спільного множника для двох чисел. Всі елементи мають однаковий знак і ціле значення.
- *Опис логічної структури :*
 - Функція `main` . Задає мансимальне значення числа для запису в масив, масив та його розміри, і передає все до функції `findFrequency`. Схема алгоритму функції:



- Функція `findGCF` . Знаходить частоту зустрічання певного елементу в масиві. Параметри: `a` - перше число; `b` - друге число; `div` - НСД. Схема алгоритму функції:



- Структура програми:

```
.
├── doc
│   ├── assets
│   ├── Lab11.md
│   └── Lab11.pdf
├── Doxyfile
├── Makefile
├── task01
│   └── src
│       ├── lib.c
│       ├── lib.h
│       └── main.c
```

- *Важливі елементи програми:*

- Формування змінних. Масив з випадковими числами.

```
#define SIZE 10
int max = 100;
int a[SIZE];
for (int i = 0; i < size; i++) {
    arr[i] = rand() % max;
}
```

- Виконання алгоритму визначення частоти повторюваності елементів.

```
int cor = 0;
for (int i = 0; i < size; i++) {
    for (int k = 0; k < size; k++) {
        for (int j = i - 1; j > 0; j--){
            if (arr[j] == arr[i]) {
                i++;
                cor++;
            }
        }
        if (arr[i] != arr[k]) {
            *(res + (i - cor) * 2) = arr[i];
        }else{
            *(res + ((i - cor) * 2) + 1) += 1;
        }
    }
}
```

Варіанти використання:

З допомогою відладника ставимо точку зупинки на строках виводу функції (для визначення початкового стану змінних) та `return 0;` (для визначення стану змінних).

Виклик функції `findFrequency` з початковими даними:

Threa	Frame	Variable	Value	Type
1	0	Local Variables		
		max	100	int
		▼ a	[10]	int [10]
		0	83	int
		1	86	int
		2	77	int
		3	15	int
		4	93	int
		5	35	int
		6	86	int
		7	92	int
		8	49	int
		9	21	int
		ressize	18	int
		▶ res	0x4052a0	int *
		Function Arguments		

Після знаходження частоти зустрічання чисел:

Variable	Value	Type
▼ In scope expressions		
▼ (int[18])*(int*)res	[18]	int [18]
0	83	int
1	1	int
2	86	int
3	2	int
4	77	int
5	1	int
6	15	int
7	1	int
8	93	int
9	1	int
10	35	int
11	1	int
12	92	int
13	1	int
14	49	int
15	1	int
16	21	int
17	1	int
Out of scope expressions		
Target Terminal Context Breakpoints Registers Memory Expression Monitor		

Висновки:

Для виконання лабораторної роботи я навчився реалізовувати алгоритми функції, та використовувати показники.

