

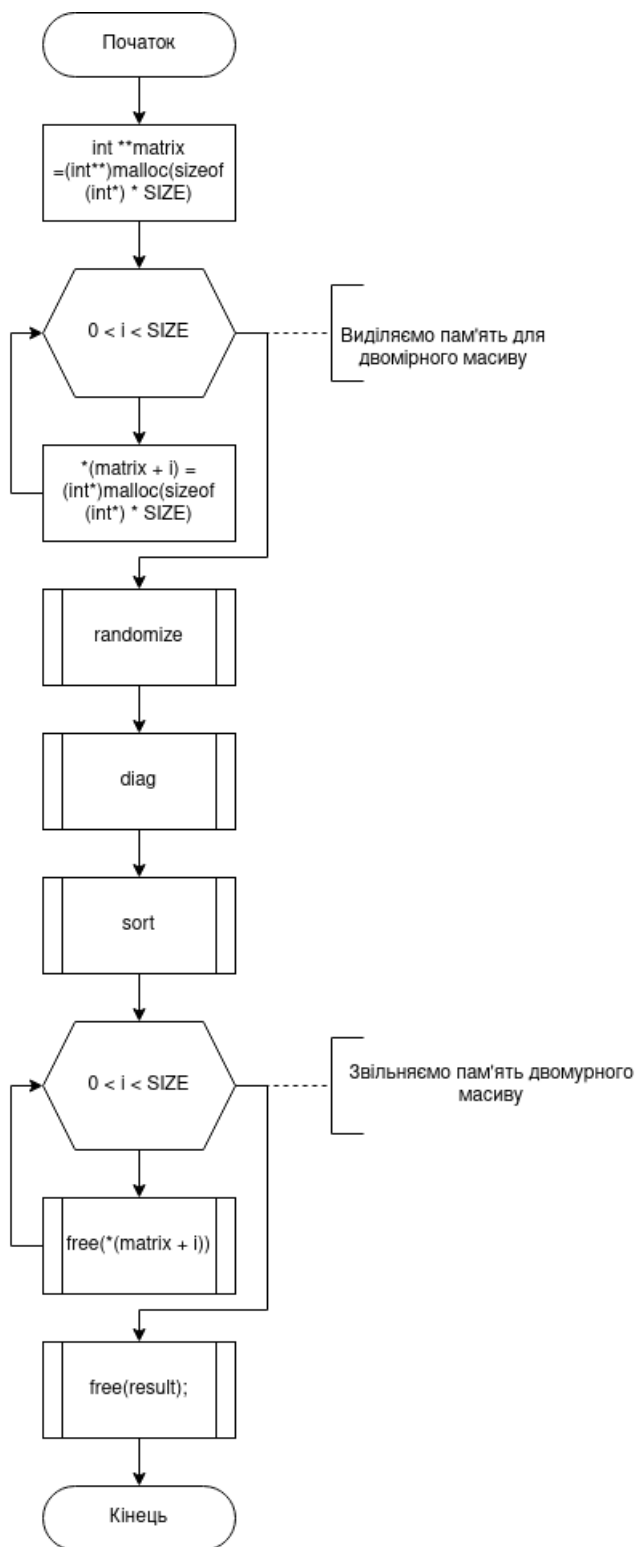
Лабораторна робота №11

Вимоги:

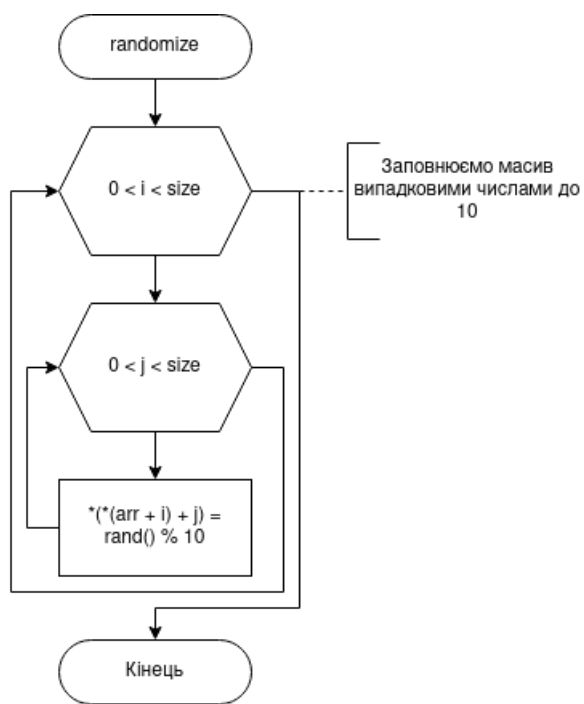
- *Розробник:* Закусілов Микита
- *Загальне завдання:* Ознайомлення із показчиками.
- *Індивідуальне завдання:* Винесення елементів головної діагоналі двомірної матриці в окремий масив та його сортування.

Опис програми:

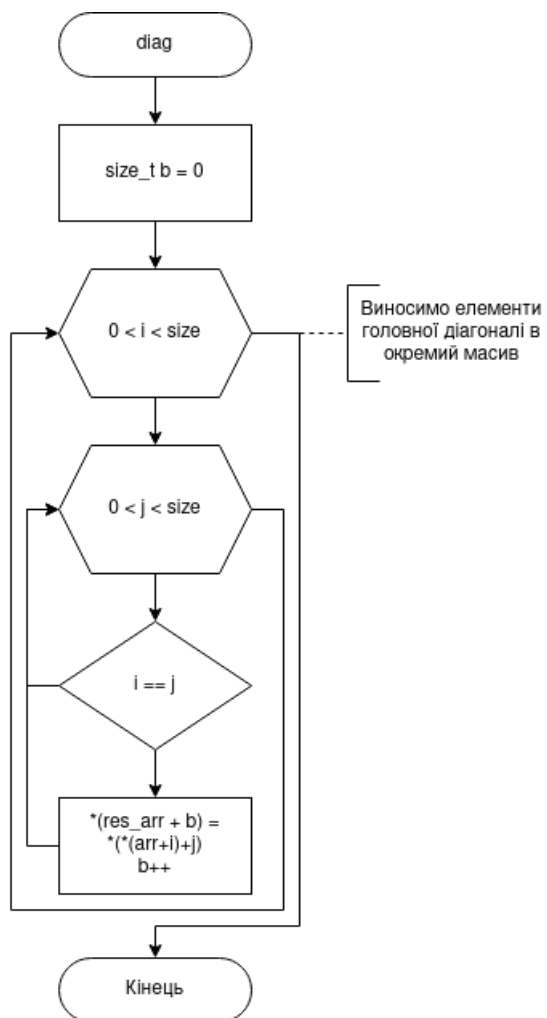
- *Функціональне призначення:* Знаходження елементів головної діагоналі.
- *Опис логічної структури.*
 - Функція `main()` .
 - Виділення динамічної пам'яті для масивів.
 - Виклик функцій: `randomize`, `diag`, `sort`.
 - Звільнення виділеної пам'яті.



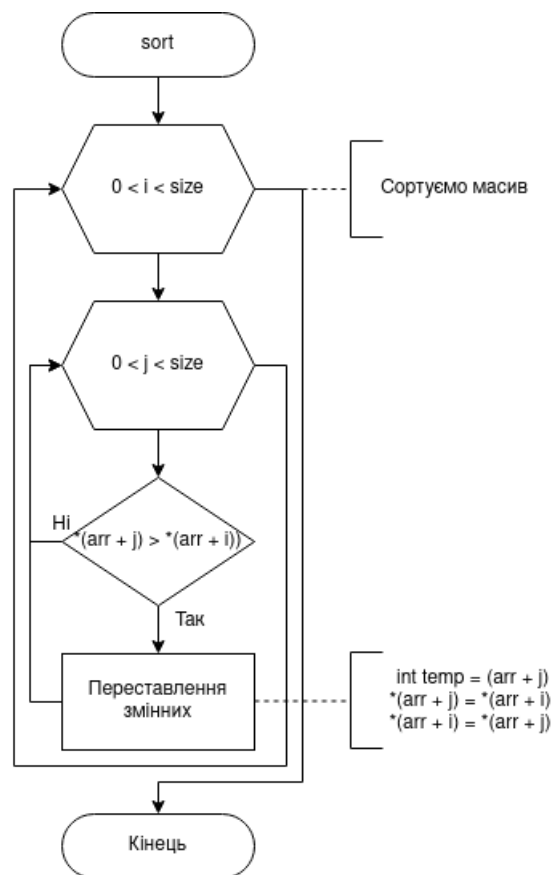
- Функція `randomize`.
 - Заповнення масиву випадковими числами.



* Функція `diag`. * Винесення елементів головної діагоналі в окремий масив.



- Функція `sort`.



- Сортування масиву методом бульбашки.

- Структура програми:

```

.
├── doc
│   ├── assets
│   │   ├── screen1.png
│   │   ├── randomize.png
│   │   ├── diag.png
│   │   ├── sort.png
│   │   └── main.png
│   ├── report.pdf
│   └── report.md
├── Doxyfile
├── Makefile
├── src
│   ├── lib.h
│   ├── lib.c
│   └── main1.c

```

- Важливі елементи програми.

- Визначення елементів головної діагоналі: `size_t b = 0; for(size_t i=0; i < size; i++){ for(size_t j=0; j < size; j++){ if(i == j){ *(res_arr + b)=*(arr+i)+j); b++; } }`
- Сортування масиву методом бульбашки: `for (size_t i = 0; i < size; i++){ for(size_t j=0; j < size;j++){ if (*(arr + j) > *(arr + i)){ int temp = *(arr + j); *(arr + j) = *(arr + i); *(arr + i) = temp; } }`

Варіанти використання програми

Ми майже не використовуємо вхідні дані, окрім розміру масиву та максимального значення функції `rand`.

Отримаємо такі вихідні дані:

▼ Выражения в области видимости		
▼ (int[3])*(int*)result	[3]	int [3]
0	3	int
1	3	int
2	9	int

Тобто елементи головних діагоналі: 3, 3, 9.

Висновок

Було створено програму для визначення, запису та сортування елементів головної діагоналі квадратної матриці.