**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
імені ІГОРЯ СІКОРСЬКОГО»**

**Інститут прикладного системного аналізу  
Кафедра системного проектування**

**Звіт**

**про виконання лабораторної роботи №10   
з дисципліни “Алгоритми та програмування”**

Виконав:  
студент I курсу, групи ДА-12  
Кракович Павло Дмитрович

Прийняв:

к.т.н., доцент Безносик О. Ю.

Київ – 2021

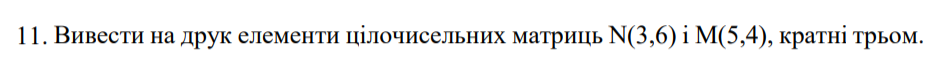
1. Скласти алгоритм і програму рішення запропонованої задачі (рис. 1)  
   

Рис. 1

1. Аналіз умови задачі.

Виходячи з умови задачі, потрібно скласти алгоритм, який буде використовувати функції для роботи з динамічним масивом.

1. Блок схема (рис. 2-5)

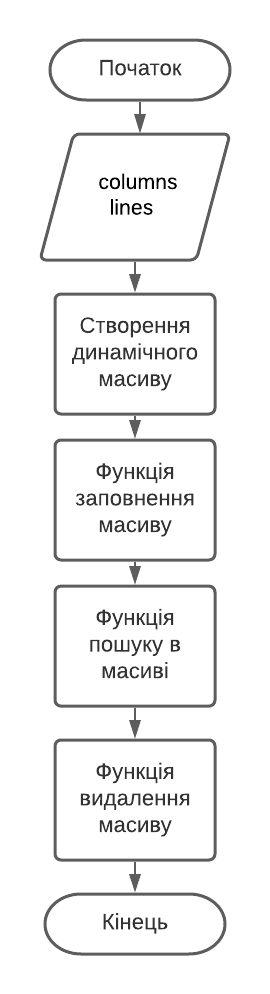


Рис. 2

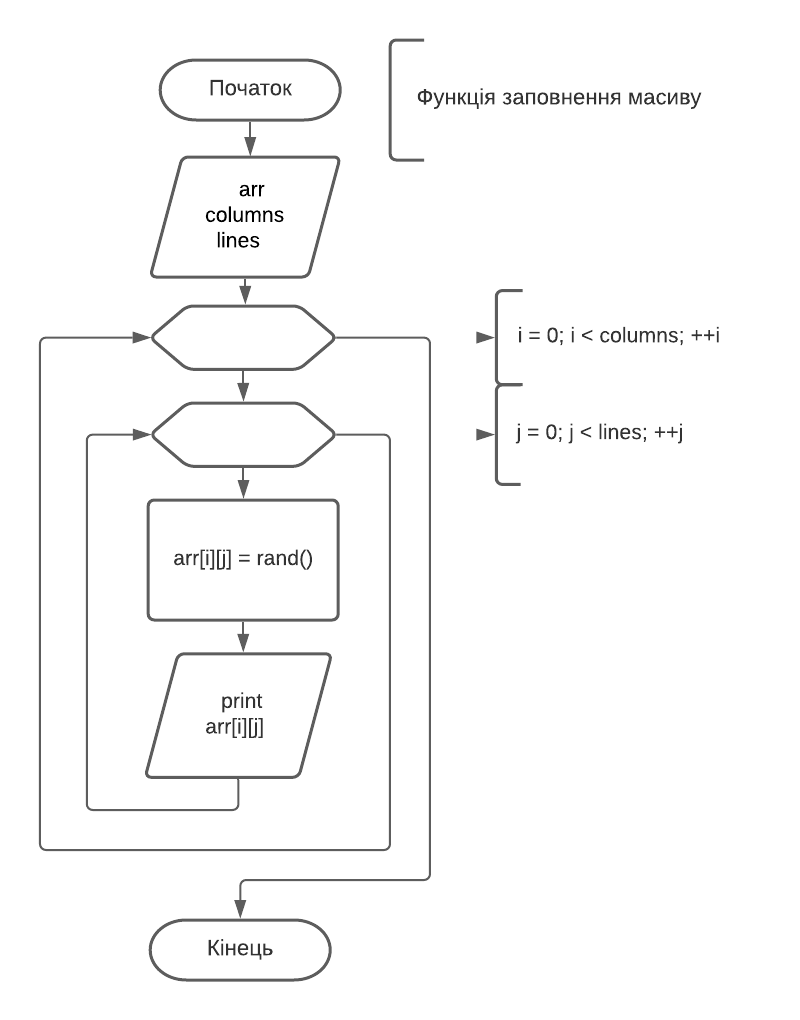


Рис. 3

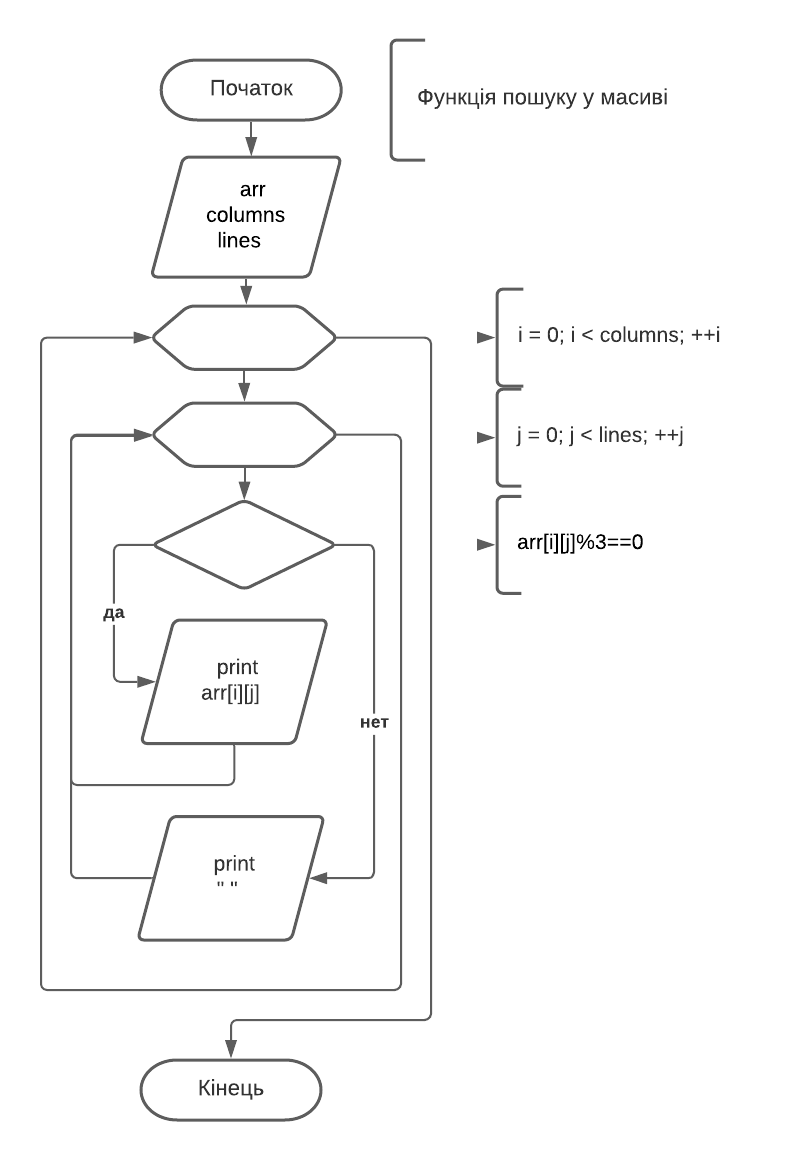


Рис. 4

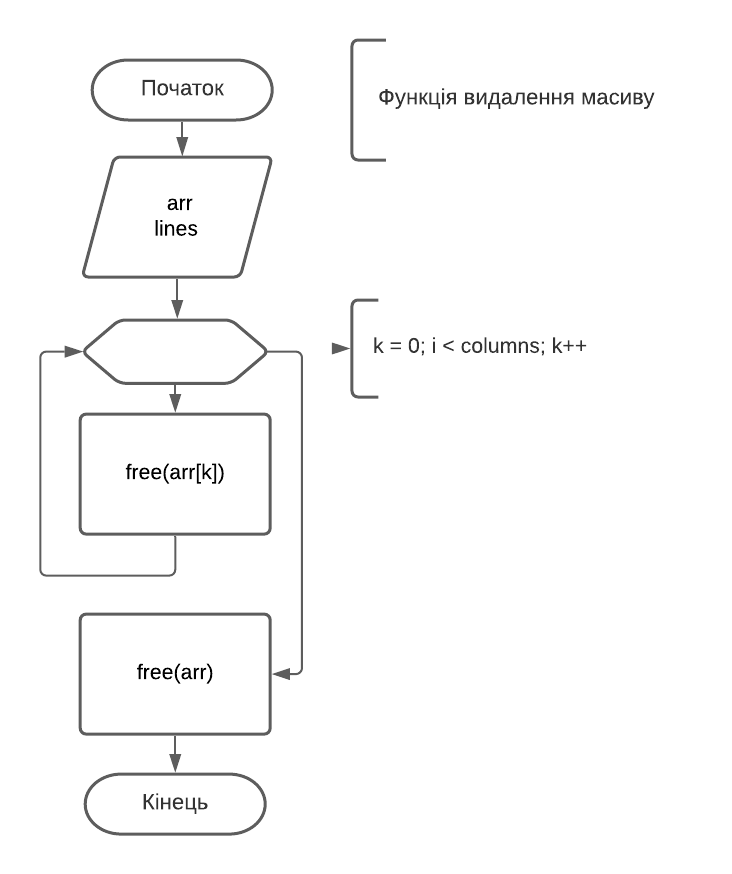


Рис. 5

1. Код (рис. 6)
2. #include <stdio.h>  
   #include <stdlib.h>  
   #include <time.h>  
     
   void fillArray(int \*\*arr, int columns, int lines) {  
    for (int i = 0; i < columns; ++i) {  
    for (int j = 0; j < lines; ++j) {  
    arr[i][j] = rand() % 9 + 1;  
    printf("%d ", arr[i][j]);  
    }  
    printf("\n");  
    }  
    printf("\n");  
   }  
   void search(int \*\*arr, int columns, int lines) {  
    for (int i = 0; i < columns; ++i) {  
    for (int j = 0; j < lines; ++j) {  
    if (arr[i][j]%3==0) {  
    printf("%d ", arr[i][j]);  
    }  
    else {  
    printf(" ");  
    }  
    }  
    printf("\n");  
    }  
   }  
   void deleteArr(int \*\*arr, int lines) {  
    for(int k = 0; k < lines; k++) {  
    free(arr[k]);  
    }  
    free(arr);  
   }  
   int main() {  
    srand(time(NULL));  
    int columns,lines;  
    printf("Type number of elements in the first matrix:");  
    scanf("%d %d",&columns, &lines);  
     
    int \*\*arr = (int \*\*)malloc(lines\*sizeof(int \*));  
    for(int i = 0; i < lines; i++) {  
    arr[i] = (int \*)malloc(columns\*sizeof(int));  
    }  
     
    fillArray(arr,columns,lines);  
    search(arr,columns,lines);  
     
    printf("\nType number of elements in the second matrix:");  
    scanf("%d %d",&columns, &lines);  
     
    \*\*arr = (int \*\*)malloc(lines\*sizeof(int \*));  
    for(int i = 0; i < lines; i++) {  
    arr[i] = (int \*)malloc(columns\*sizeof(int));  
    }  
    fillArray(arr,columns,lines);  
    search(arr,columns,lines);  
    deleteArr(arr,lines);  
    return 0;

Рис. 6

1. Результат роботи програми: (рис. 7)

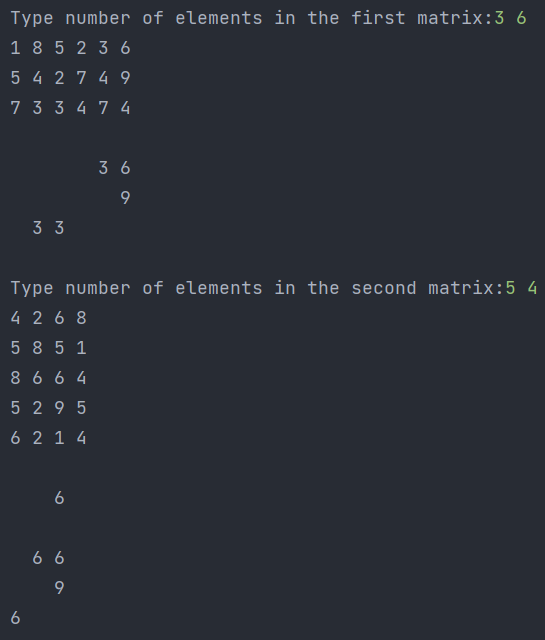


Рис. 7

Висновок

В ході виконання лабораторної роботи №10 мною було розроблено алгоритм розв’язку завдання.

Я створив динамічний масив, з яким працював завдяки функціям. Також я застосував функцію free() з бібліотеки C. Вона звільняє пам’ять, попередньо виділену викликом calloc, malloc або realloc.