

# Звіт

з алгоритмізації та програмування  
до лабораторної роботи №10

студента І-курсу КН-108

Гірника Юрія

# Звіт

## 1. Постановка та варіант завдання:

## 2. Постановка завдання

Написати програму, у якій створюються динамічні масиви й виконати їхню обробку у відповідності до свого варіанту.

7. Сформувати двовимірний масив. Знищити и з нього стовпець із заданим номером;

## 3. Текст програми та функцій:

```
#include <stdio.h>
#include <malloc.h>
#include <stdlib.h>
#include <time.h>

int i,j;

int ** create(int SIZE1, int SIZE2); //створення та заповнення масиву
void print(int **array, int SIZE1, int SIZE2); //друк створеного масиву/знищення масивів
void kick (int ** array, int del,int SIZE1, int SIZE2); //видалення заданого рядка

int main()
{
    int SIZE1, SIZE2, del;
    scanf("%d %d", &SIZE1,&SIZE2); //ввід розміру масиву
    int ** array;
    array = create(SIZE1,SIZE2); //створення та заповнення
    print(array, SIZE1, SIZE2); //друк
    printf("Delete a column with number(1-%d): ", SIZE2);
    scanf("%d", &del); //задання номера стовбця для видалення
    del = del-1;
    kick(array, del, SIZE1, SIZE2); //видалення та друк фінального варіанту
}

void print(int **array, int SIZE1, int SIZE2)
{
    for (i = 0; i<SIZE1; i++)
    {
        for(j = 0; j<SIZE2; j++)
            printf("%2d ", array[i][j]);
        printf("\n");
    } //вивід на екран
}
```

```

int** create(int SIZE1, int SIZE2)
{
    srand((int) time (NULL));
    int **array;
    array = (int **) malloc(SIZE1 * sizeof(int*));
    for (i = 0; i<SIZE1; i++)
    {
        array[i] = (int *) malloc(SIZE2 * sizeof(int));
    }//виділення потрібної пам'яті
    for (i = 0; i<SIZE1; i++)
        for(j = 0; j<SIZE2; j++)
        {
            array[i][j] = rand()%27;//заповнення випадковими числами
        }
    return array;
}

```

```

void kick (int ** array, int del, int SIZE1, int SIZE2)
{
    int **array_upd;
    int o = 0;
    array_upd = (int **) malloc(SIZE1 * sizeof(int*));
    for (i = 0; i<SIZE1; i++)
        array_upd[i] = (int *) malloc((SIZE2-1) * sizeof(int));
    //виділення пам'яті
    for (i = 0; i<SIZE1; i++)
        for(j = 0, o = 0; j<SIZE2-1; j++)
        {
            if (o>SIZE2-1) break;
            if (o==del) o++;
            array_upd[i][j] = array[i][o];
            o++;
        }//заповнення
    for (i = 0; i<SIZE1; i++)
    {
        for(j = 0; j<SIZE2-1; j++)
        {
            printf("%2d ", array_upd[i][j]);
        }
        printf("\n");
    }//друк
    for (i = 0; i<SIZE1; i++)
    {
        free(array[i]); //знищення масивів
        free(array_upd[i]);
    }
}

```

### 3. Результати роботи програми:

```
~/workspace/ $ ./10
4 5
13 4 5 6 5
26 1 10 11 3
3 10 7 9 5
5 8 11 24 4
Delete a column with number(1-5): 4
13 4 5 5
26 1 10 3
3 10 7 5
5 8 11 4
~/workspace/ $ ./10
3 3
15 1 22
19 22 21
15 15 22
Delete a column with number(1-3): 1
1 22
22 21
15 22
~/workspace/ $ ./10
5 4
25 17 7 7
14 11 5 6
11 11 18 9
13 20 18 9
4 9 4 5
Delete a column with number(1-4): 2
25 7 7
14 5 6
11 18 9
13 18 9
4 4 5
```

