

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота
з дисципліни
«Алгоритмізація та програмування»

Виконав:
студент групи КН-109
Кульчицька Олена
Викладач:
Варецький Я.Ю.

Львів – 2018 р.

Лабораторна робота № 10

Тема: "Динамічні масиви"

Мета: Організація динамічних масивів.

2. Постановка завдання

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

3. Порядок виконання роботи

1. Ввести розмір масиву;
2. Сформувати масив за допомогою операції new або бібліотечних функцій malloc (calloc);
3. Заповнити масив (можна за допомогою датчика випадкових чисел);
4. Виконати завдання варіанту, сформувати новий масив(и) – результат(и);
5. Надрукувати масив(и) – результат(и);
6. Знищити динамічні масиви за допомогою операції delete або бібліотечної функції free.

Варіант 15

Сформувати двовимірний масив. Знищити з нього рядок і стовпець, на перетині яких перебуває максимальний елемент.

```
#include <stdio.h>
#include <stdlib.h>
int **make_mas(int rows,int columns)
{
    int **mas;
    mas=(int**)malloc(rows*sizeof(int));
    for(int i=0;i<rows;i++){
        mas[i]=(int*)malloc(columns*sizeof(int));
    }
    for(int i=0;i<rows;i++){
        for(int j=0; j<columns;j++){
            mas[i][j]=rand()%200;
        }
    }
    return mas;
}
int main()
{
```

```

int max=0;
int max_r,max_c;
int rows,columns;
printf("Enter rows:\n");
scanf("%d",&rows);
printf("Enter columns:\n");
scanf("%d",&columns);
int **array;
array=make_mas(rows, columns);

for(int i=0;i<rows;i++){
    for(int j=0; j<columns;j++){
        printf("%d\t",array[i][j]);
    }
    printf("\n");
}
for(int i=0;i<rows;i++){
    for(int j=0; j<columns;j++){
        if(max<array[i][j]) {
            max=array[i][j];
            max_r = i;
            max_c = j;
        }
    }
}
printf("%d\n",max);
for(int i=0;i<rows;i++){
    for(int j=0;j<columns;j++){
        if(max_r==i||max_c==j){
            array[i][j]=-1;
        }
    }
}

for(int i=0;i<rows;i++){
    for(int j=0; j<columns-1;j++){
        if(array[i][j]==-1){
            array[i][j]= array[i][j+1];
            array[i][j+1] = -1;
        }
    }
}

```

```

for(int i=0;i<rows-1;i++){
    for(int j=0; j<columns;j++){
        if(array[i][j]==-1){
            array[i][j]= array[i+1][j];
            array[i+1][j] = -1;
        }
    }
}
printf("\n");
for(int i=0;i<rows;i++){
    for(int j=0; j<columns;j++){
        if(array[i][j] != -1){
            printf("%d\t",array[i][j]);
        }
    }
    printf("\n");
}
}
}

```

```

Enter rows:
5
Enter columns:
6
41      67      134      100      169      124
78      158      162      64      105      145
81      27      161      91      195      142
27      36      191      4      102      153
92      182      21      116      118      95
Max element = 195

41      67      134      100      124
78      158      162      64      145
27      36      191      4      153
92      182      21      116      95

Process finished with exit code 0

```