

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

**Кафедра систем штучного інтелекту**



**Лабораторна робота №10**

**з дисципліни**

**«Алгоритмізація та програмування»**

**Виконав:  
студент групи КН-110  
Гавриляк Тарас**

**Львів – 2018 р.**

## Лабораторна робота № 10.

**Тема: "Динамічні масиви"**

**Мета:** Організація динамічних масивів.

### Постановка завдання

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

Сформувати двовимірний масив. Знищити з нього рядок із заданим номером;

Код:

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

```
int** make();
```

```
int** del();
```

```
int main (void)
```

```
{
```

```
    int n, m;
```

```
    srand(time(NULL));
```

```
    printf("input size of matrix: ");
```

```
    scanf("%d", &n);
```

```
    printf("\n");
```

```
int** mtx = make(n);
```

```
printf("\nchoose what row you want delete: ");//g
```

```
scanf ("%d", &m);
```

```
printf("\n");
```

```
del(m, mtx, n);
```

```
for (int f = 0; f<n-1; f++)
```

```
    free(mtx[f]);
```

```
free (mtx);
```

```
return 0;
```

```
}
```

```
int** make(int n)
```

```
{
```

```
    if(n < 0)
```

```
        return NULL;
```

```
    int** mas;
```

```
    int i, j;
```

```
    mas = (int**) malloc(n * sizeof(int*));
```

```

for (j=0; j<n; j++)
{
    mas[j] = (int*) malloc(n * sizeof(int));

    for(i = 0; i<n; i++)
    {
        mas[j][i] = rand()%10;
    }
}

for (i=0; i<n; i++)
{
    for (j=0; j<n; j++)
    {
        printf ("%d ", mas[i][j]);
    }
    printf("\n");
}

return mas;
}

```

```

int** del(int row, int**matrix, int n)
{
    int j, i;

    for (i = 0; i < n ; i++)

```

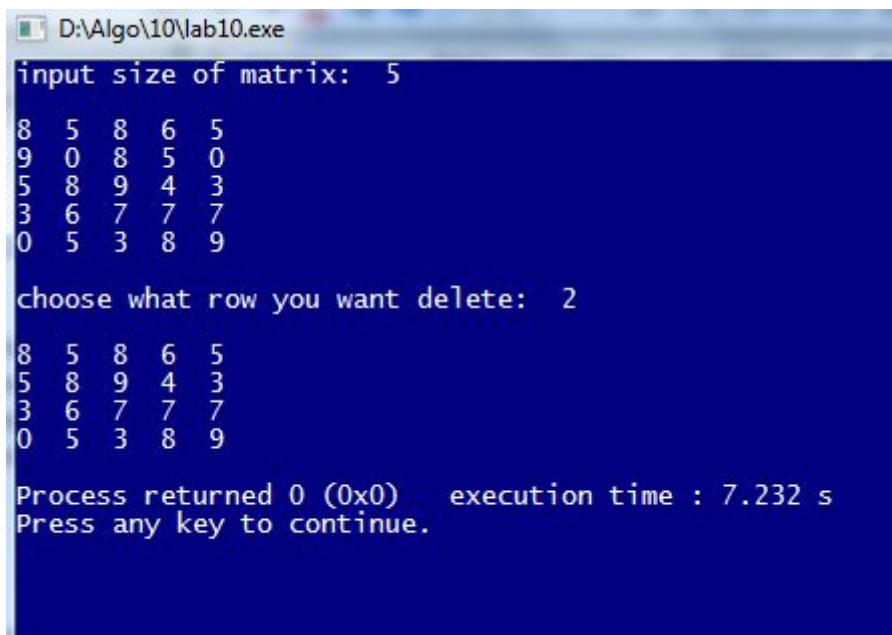
```

{
    if (matrix[i]!=matrix[row-1])
    {
        for (j=0; j<n; j++)
        {
            printf ("%d ", matrix[i][j]);
        }
        printf("\n");
    }
}

return matrix;
}

```

Результат:



```

D:\Algo\10\lab10.exe
input size of matrix: 5
8 5 8 6 5
9 0 8 5 0
5 8 9 4 3
3 6 7 7 7
0 5 3 8 9

choose what row you want delete: 2
8 5 8 6 5
5 8 9 4 3
3 6 7 7 7
0 5 3 8 9

Process returned 0 (0x0)   execution time : 7.232 s
Press any key to continue.

```

