

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА
ПОЛІТЕХНІКА»

Кафедра систем штучного інтелекту



Лабораторна робота №10
З курсу “Алгоритмізація та програмування”

Виконала:
ст.гр. КН-110
Бурцьо Ольга
Викладач:
Гасько Р.Т.

Львів – 2018

Лабораторна робота №10

Варіант №5

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.

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.

Сформувати одновимірний масив. Знищити з нього К елементів, починаючи із заданого номера, додати К елементів, починаючи із заданого номера;

Код програми:

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include<time.h>
4
5 int* delete_el(int * mas, int numb, int kill, int start);
6 int* add_el(int * mas, int numb, int revive, int start_new);
7
8 int main(void)
9 {
10     int n, i;
11     int *mas;
12
13     printf("Enter number of elements: ");
14     scanf("%d", &n);
15
16     srand(time(0));
17
18     mas=(int*)malloc(n * sizeof(int));
19
20     printf("Your massive: [ ");
21     for(i=0; i<n; i++)
22     {
23         mas[i]=rand()%100;
24         printf("%d, ", mas[i]);
25     }
26     printf("]\n");
27
28     int st;
29     int kill;
30     printf("Input start killing point: ");
31     scanf("%d", &st);
32     printf("Input how many victims you want have: ");
33     scanf("%d", &kill);
34
35     int * new_mas = delete_el(mas, n, kill, st);
36     n = n - kill;
37     printf("Your nem massive: [");
38     for(i=0; i< n; i++)
39     {
40         printf("%d, ", new_mas[i]);

```

```

41     }
42     printf("]\n");
43
44     int start_new;
45     printf("Input new start point: ");
46     scanf("%d", &start_new);
47
48     int * revived_mas = add_el(new_mas, n, kill, start_new);
49     n = n + kill;
50     printf("Your exciting massive: [");
51     for(i=0;i<n;i++)
52     {
53         printf("%d, ", revived_mas[i]);
54     }
55     printf("]\n");
56
57 }
58
59 int* delete_el(int * mas, int numb, int kill, int start)
60 {
61     start--;
62     int * new_mas=(int*)malloc(sizeof(int) * (numb - kill));
63     int i,j;
64     for(i=0; i< start; i++)
65     {
66         new_mas[i]=mas[i];
67     }
68     for(j=start + kill; j < numb; j++,i++)
69     {
70         new_mas[i]=mas[j];
71     }
72     return new_mas;
73 }
74
75 int* add_el(int * mas, int numb, int revive, int start_new)
76 {
77     int * revived_mas=(int*)malloc(sizeof(int) * (numb + revive));
78     int i,j;

```

```

79     for(i=0; i< start_new; i++)
80     {
81         revived_mas[i]=mas[i];
82     }
83     for(i = start_new; i < (start_new + revive); i++)
84     {
85         revived_mas[i] = rand()%100;
86     }
87     for(j=start_new; j < (start_new + revive); j++,i++)
88     {
89         revived_mas[i]=mas[j];
90     }
91     return revived_mas;
92 }

```

Результат:

```

Enter number of elements: 8
Your massive:[ 50, 63, 36, 24, 38, 92, 59, 54, ]
Input start killing point: 2
Input how many victims you want have: 4
Your nem massive: [50, 92, 59, 54, ]
Input new start point: 3
Your exciting massive: [50, 92, 59, 79, 60, 5, 53, 54, ]

```