

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

Кафедра автоматизированных систем управления (АСУ)

СТАТИЧЕСКИЕ И ДИНАМИЧЕСКИЕ МАССИВЫ В СИ

Отчёт о лабораторных работах № 5-6 по дисциплине
«Программирование»

Студент гр. 430-2

_____ Д.А.Кондратьев

« ____ » _____ 2020г.

Руководитель

Ассистент каф. АСУ

_____ А.Е.Косова

« ____ » _____ 2020г.

Томск 2020

Оглавление

1 Цели и задачи работы.....	3
1.1 Тема.....	3
1.2 Цели.....	3
1.3 Задачи.....	3
1.4Индивидуальное задание.....	3
2 Алгоритм.....	4
3 Код программы.....	11
4 Изменение параметров программы.....	20
6 Выводы.....	25

1 Цели и задачи работы

1.1 Тема

- Статические и динамические массивы в языке си

1.2 Цели

- **Освоить методы создания и обработки статических/динамических массивов**

1.3 Задачи

- Научится создавать функции для обработки массивов
- Обучится работе с памятью

1.4 Индивидуальное задание

В матрице A размера $n \times n$ определить номер столбца, сумма элементов которого максимальна. Если этот номер больше трех, то поменять местами все элементы главной и побочной диагоналей матрицы.

2 Алгоритм

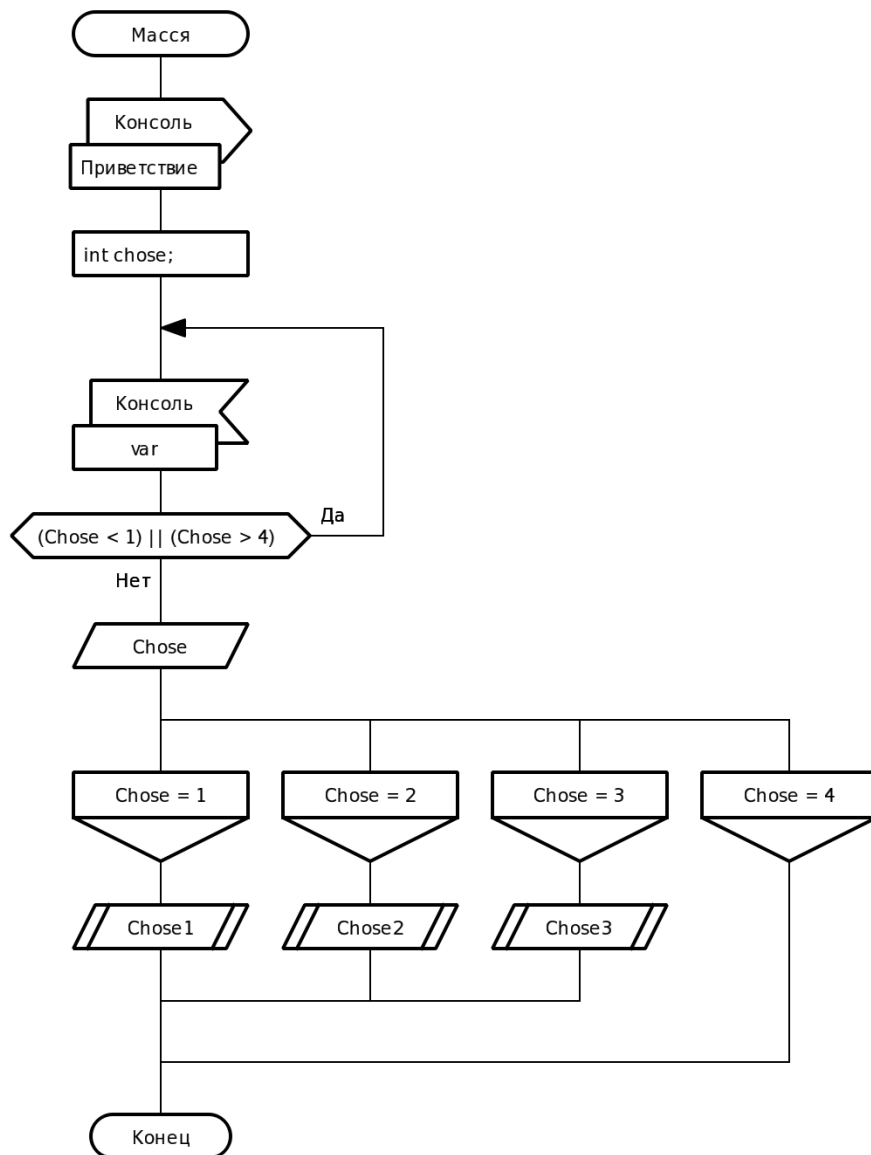


Рис. 1 main

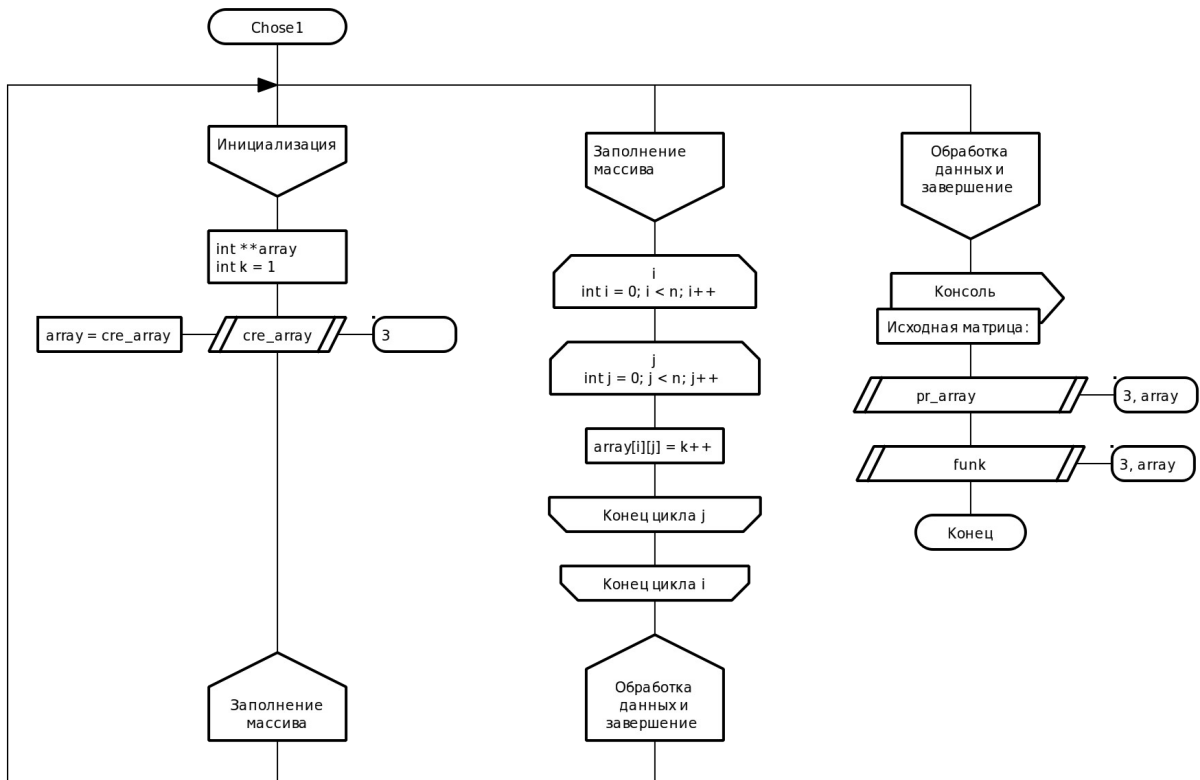


Рис. 2 chose1

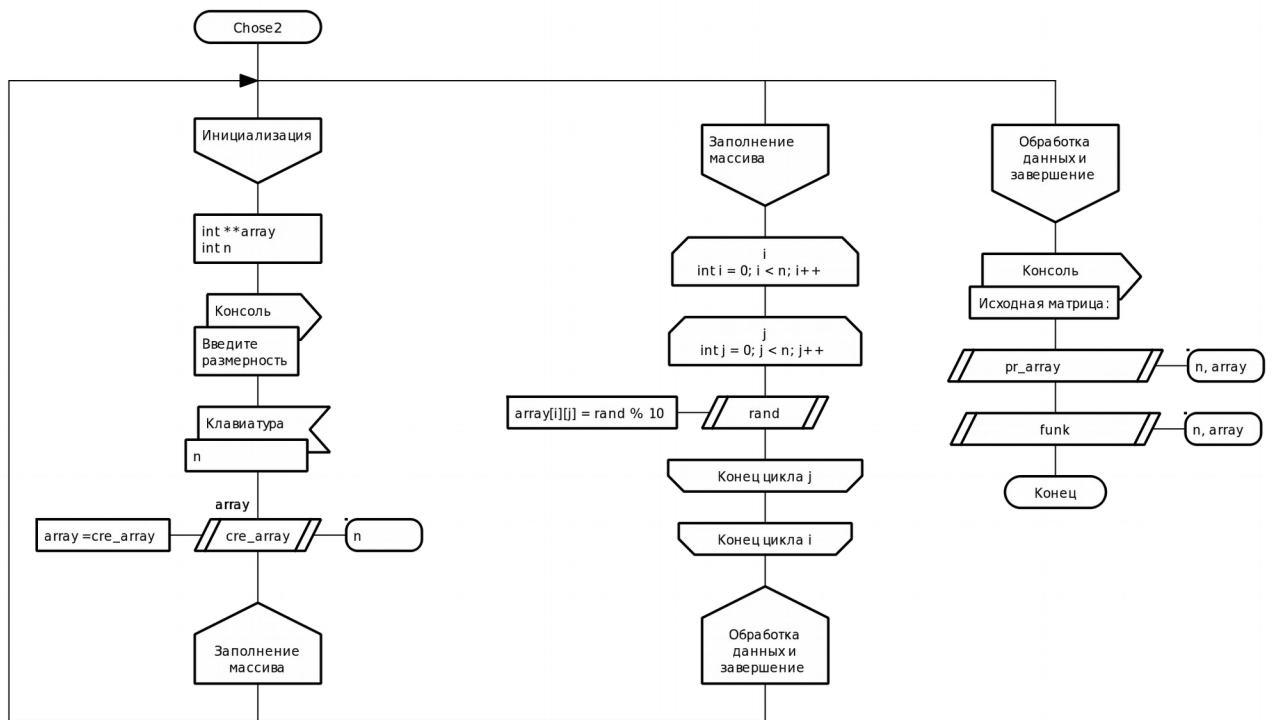


Рис. 3 chose2

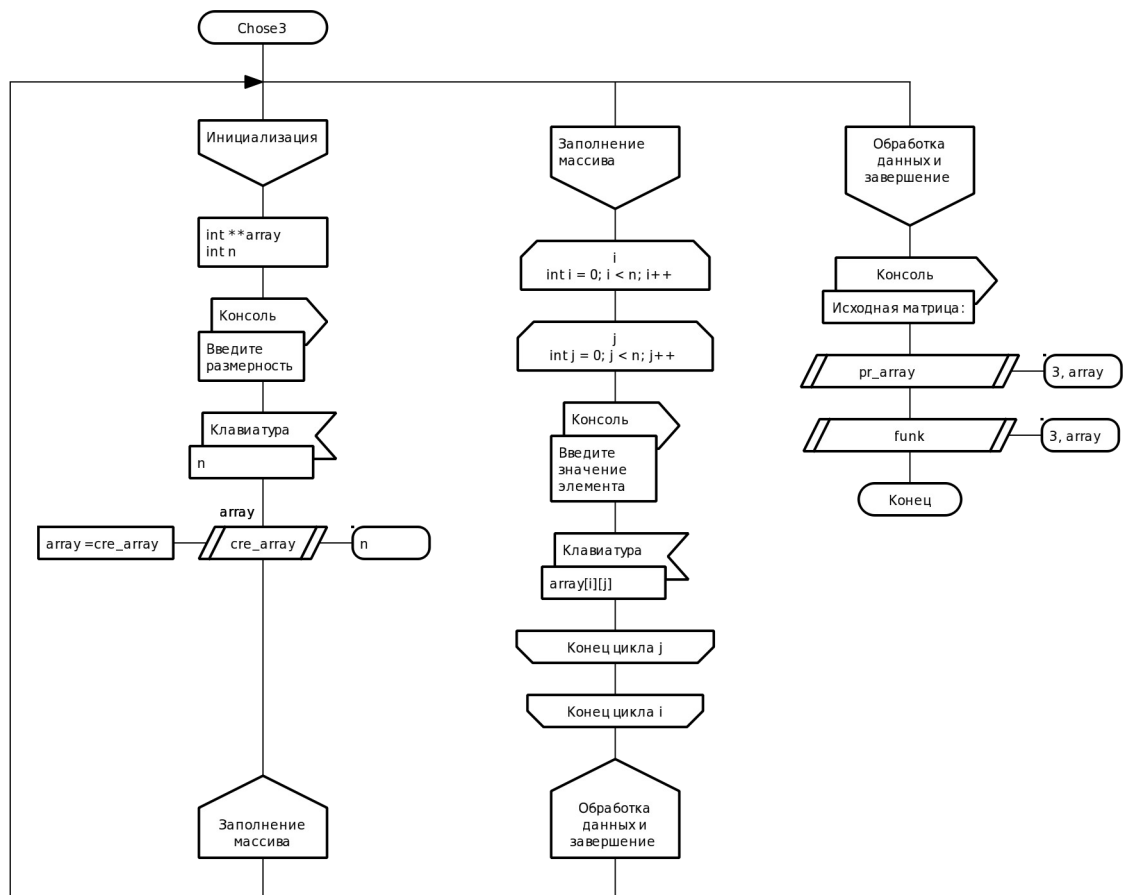


Рис. 4 chose3

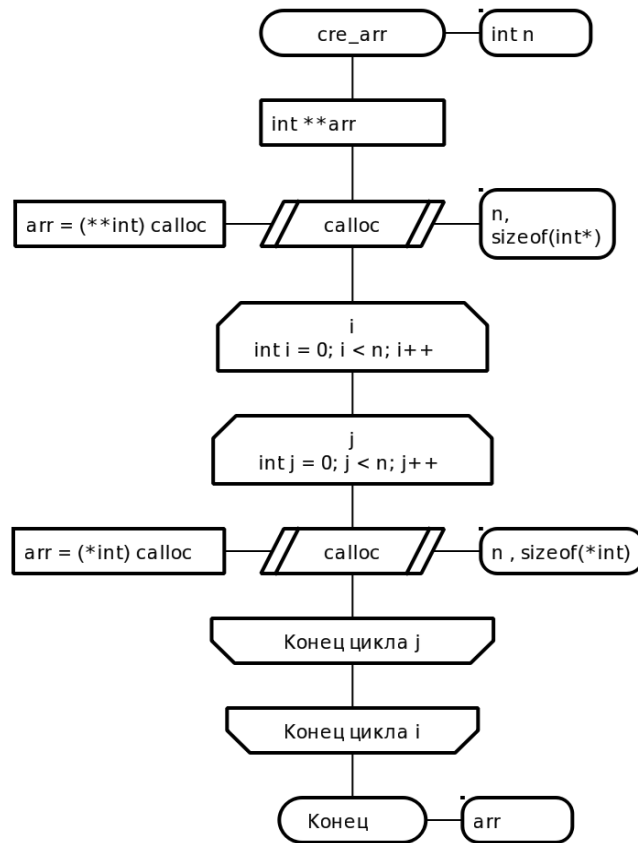


Рис. 5

функция создания динам. массива

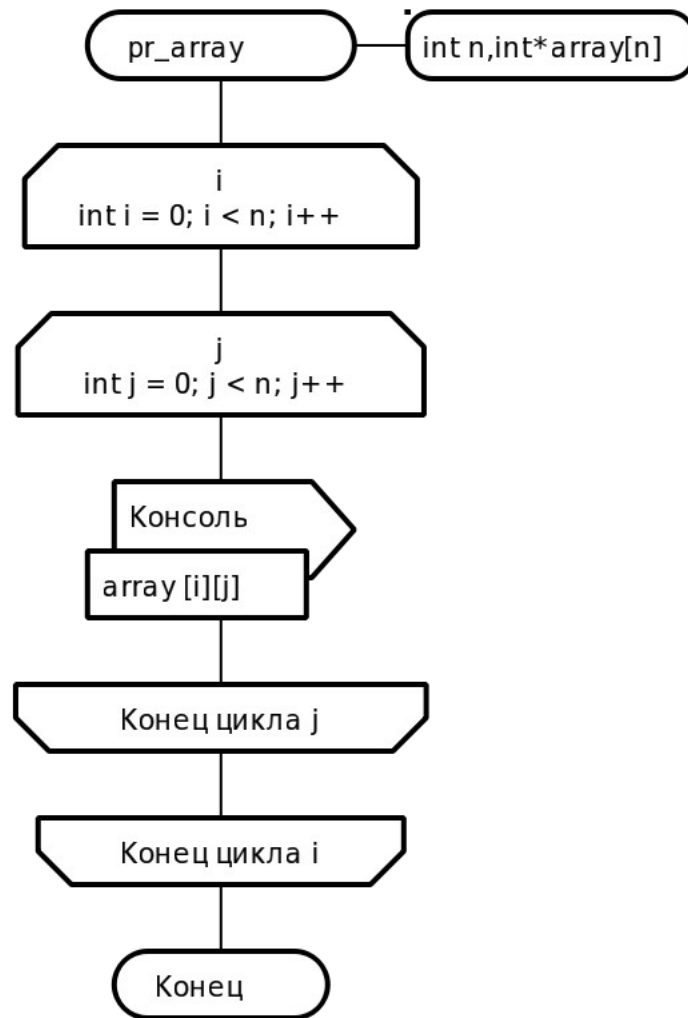


Рис. 6

функция вывода динамического массива

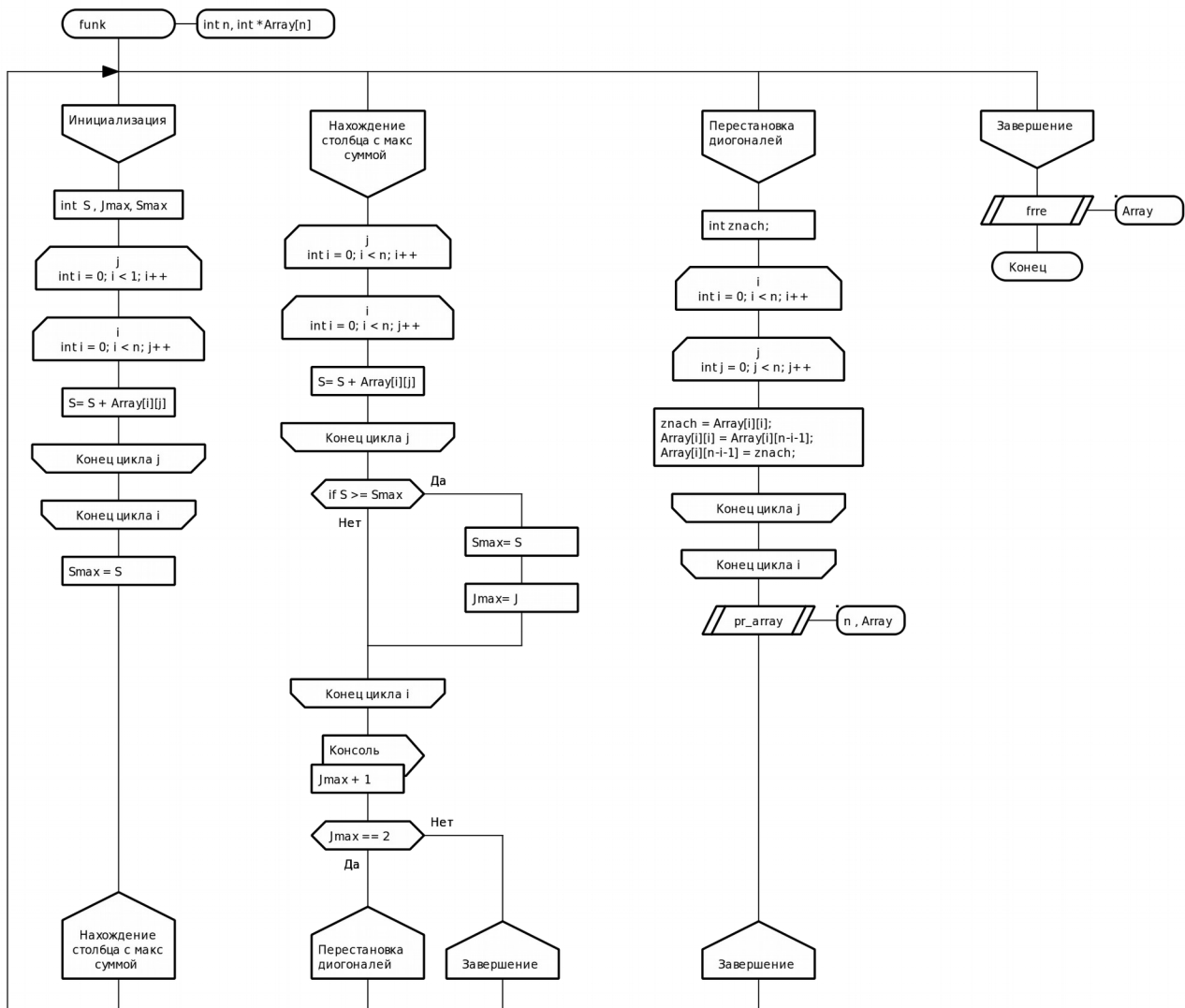


Рис. 7

функция обработки массива

3 Код программы

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```
#include<time>
```

```
/* В матрице A размера n*n определить номер столбца, сумма элементов которо-
го максимальна. Если этот номер больше трех, то поменять местами все
элементы
```

```
главной и побочной диагоналей матрицы.*/
```

```
#define h 3
```

```
void funk(int n, int *Array[n])
```

```
{
```

```
    int S= 0,Jmax = 0,Smax;
    присваиваем Smax
```

```
// находим сумму 1 столбца и
```

```
    // что бы было с чем сравнивать в
    дальнейшем
```

```
    for(int j = 0; j< 1;j++)
```

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

```
            S = S + Array[i][j];
```

```
    Smax = S;
```

```
    for(int j =0; j<n;j++)
    столбец с макс суммой
```

```
// Находим максимальную
```

```
{
```

```
    S = 0;
```

```
for(int i=0; i<n; i++)
    S = S + Array[i][j];
```

```
if (S >= Smax)
{
    Jmax = j;
    Smax = S;
}
```

```
}
```

```
printf("\nОтвет:%d\n",Jmax + 1);           // Выводим номер столбца с
макс суммой(при этом нумерацию считаем с 1)
```

```
if (Jmax == 2)                             //Если сумма 3 столбца
наибольшая то меняем гл и побоч диагонали
```

```
{
    printf("Новая матрица A:\n");
    int znach;
    for(int i = 0 , j =n -1 ; i<n;i++, j--)
    {
        znach = Array[i][i];
        Array[i][i] = Array[i][n-i-1];
        Array[i][n-i-1] = znach;
    }
```

```
pr_array(n,Array);           //Вывод новой матрицы
```

```

    }

    free(Array);           // Освобождаем память
}

void pr_array_stat();

void funk_stat(int n, int Array[][n])
{
    int S= 0,Jmax = 0,Smax;           // находим сумму 1 столбца и
    присваиваем Smax                  // что бы было с чем сравнивать в
                                      // дальнейшем
    for(int j = 0; j< 1;j++)
        for(int i = 0; i < n; i++)
            S = S + Array[i][j];
    Smax = S;

    for(int j =0; j<n;j++)           // Находим максимальную
    столбец с макс суммой
    {
        S = 0;
        for(int i=0; i<n; i++)
            S = S + Array[i][j];

        if (S >= Smax)
        {
            Jmax = j;
            Smax = S;
        }
    }
}

```

```
}
```

```
}
```

```
printf("\nОтвет:%d\n",Jmax + 1);          // Выводим номер столбца с
макс суммой(при этом нумерацию считаем с 1)
```

```
if (Jmax == 2)                            //Если сумма 3 столбца
наибольшая то меняем гл и побоч диагонали
```

```
{
```

```
printf("Новая матрица A:\n");
```

```
int znach;
```

```
for(int i = 0 , j =n -1 ; i<n;i++, j--)
```

```
{
```

```
    znach = Array[i][i];
```

```
    Array[i][i] = Array[i][n-i-1];
```

```
    Array[i][n-i-1] = znach;
```

```
}
```

```
pr_array_stat(n,Array);                  //Вывод новой матрицы
```

```
}
```

```
}
```

```
void pr_array(int n,int *array[n])
```

```
{
```

```

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

```

```

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

```

```

int **cre_arr(int n)
{
    int **arr;

    arr = (int**)calloc(n,sizeof(int**)); //создаём массив указателей на указатели
    for(int i = 0; i<n; i++)

```

```

arr[i] = (int*)calloc(n,sizeof(int*)); // заполняем 1 массив указателями
if( !arr ) return NULL;

```

```

return arr;

```

```

}

```

```

void chose1()

```

```

{

```

```

    int k = 1;

```

```

    int array[h][h];

```

```

    for(int i =0; i<h;i++)

```

```

        for(int j=0; j<h; j++)

```

```

            array[i][j] = k++ ;

```

```

    printf("Исходная матрица:\n");

```

```

    pr_array_stat(h, array);

```

```

    funk_stat(h, array);

```

```

    printf("\nТааакс, на этом всё, если понадобится запустите снова, хорошо
=)?\n");

```

```

}

```

```

void chose2()

```



```

{
    int n;
    int **array;
    srand(time(NULL));

    printf("Очень хорошо, в таком случае введите размерность квадратной
матрицы:\n");
    scanf("%d",&n);

    array = cre_arr(n);

    for(int i=0; i<n;i++)
        for(int j=0; j<n; j++)
            array[i][j] = rand() % 10;

    printf("исходная матрица:\n");
    pr_array(n, array );
    funk(n,array);

    printf("\nФуууух вроде всё, ну вы это запускайте если что :)\n");

}

void chose3()
{
    int n;

```

```

int **array;

srand(time(NULL));

printf("Угу, тогда скорее начнём! Введите размерность квадратной
матрицы:\n");

scanf("%d",&n);

array = cre_arr(n);

for(int i =0; i<n;i++)
    for(int j=0; j<n; j++)
        {
            printf("а сейчас введите элемент A[%d%s%d%s\n",i+1,"]
["j+1,"]");
            scanf("%d",&array[i][j]);
        }

printf("исходная матрица:\n");
pr_array(n, array );
funk(n,array);

printf("\nЭто всё, если что я буду на вашем электронном носителе :~).\n");

}

int main()

```

```
{
```

```
int chose = 8;
```

```
printf("Привет я Масся, программа которая умеет в матрице натуральных чисел  
А (размера n*n) определить номер столбца, сумма элементов которого  
максимальна(Беру наибольший номер максимального столбца из равных)!\n\
```

```
При том если этот номер больше трех, то я меняю местами все элементы  
главной и побочной диагоналей матрицы!\n\n");
```

```
printf("А сейчас выберете один из предложенных вариантов:\n\
```

```
1:Я сама создам и заполню массив.\n\
```

```
2:Вы введёте размер, а заполнение оставите мне.\n\
```

```
3:Вы введёте размер и заполните массив.\n\
```

```
4:Выход из программы \n");
```

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

```
while ( (chose < 1) || (chose > 4) )
```

```
{
```

```
    printf("Некорректное значение, пожалуйста, повторите попытку\n");
```

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

```
}
```

```
if (chose == 1)chose1();
```

```
else if (chose == 2)chose2();
```

```
else if (chose == 3)chose3();
```

```
else if (chosed == 4)printf("Выход из программы...приятного времени суток !!;\n");
```

```
return 0;
```

```
}
```

4 Изменение параметров программы

4.1 Изменив параметры циклов так, чтобы обрабатываемый массив был меньше объявленного, программа обрабатывает только часть заданного массива.

4.2 Изменив параметры циклов так, чтобы обрабатываемый массив был больше объявленного, программа заходит в другие ячейки памяти,и появляются номера этих ячеек .

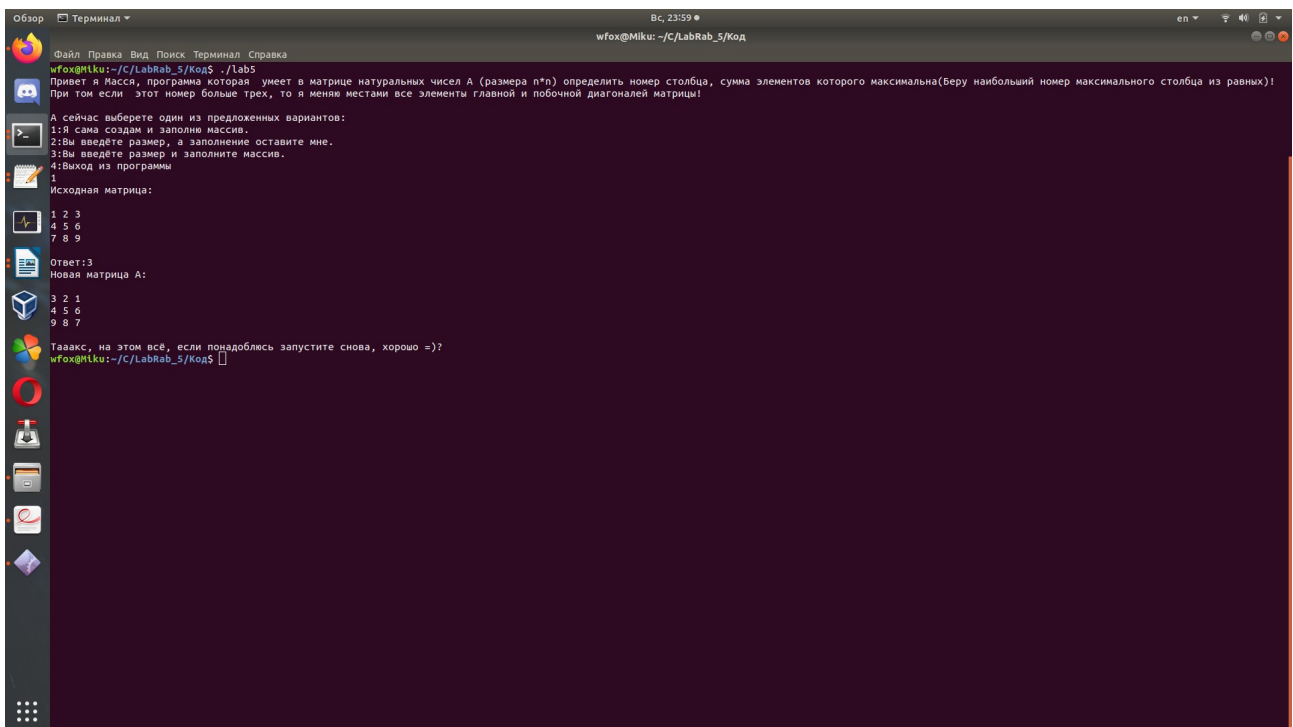
5 Тестирование программы

Тест 1

Введённые данные: Chose = 1;

Ожидаемый результат: программа по порядку заполнит массив значениями от 1 до 9 и, так как 3 столбец является наибольшим, поменяет главную и побочную диагональ;

Полученный результат:



```
Обзор Терминал
wfox@Miku: ~/C/LabRab_5/Код
wfox@Miku:~/C/LabRab_5/Код$ ./lab5
Привет я Масся, программа которая умеет в матрице натуральных чисел A (размера n*n) определить номер столбца, сумма элементов которого максимальна(беру наибольший номер максимального столбца из равных)!
При том если этот номер больше трех, то я меняю местами все элементы главной и побочной диагоналей матрицы!

А сейчас выберите один из предложенных вариантов:
1:Я сама создам и заполню массив.
2:Вы введёте размер, а заполнение оставите мне.
3:Вы введёте размер и заполните массив.
4:Выход из программы
1
Исходная матрица:
1 2 3
4 5 6
7 8 9

Ответ:3
Новая матрица A:
3 2 1
4 5 6
9 8 7

Тааакс, на этом всё, если подаблсь запустите снова, хорошо =)?
wfox@Miku:~/C/LabRab_5/Код$
```

Тест 2

Введённые данные: Chose = 2, n = 5;

Ожидаемый результат: программа создаст матрицу 5x5 и заполнит из абсолютно случайными значениями от 1 до 9 и, если 3 столбец будет наибольшим, поменяет главную и побочную диагональ;

Полученный результат:

```

Обзор Терминал
Пн, 00:02
wfox@Miku: ~/C/LabRab_5/Код

wfox@Miku:~/C/LabRab_5/Код$ ./Lab5
Привет я Масся, программа которая умеет в матрице натуральных чисел A (размера n*n) определить номер столбца, сумма элементов которого максимальна(беру наибольший номер максимального столбца из равных)!
При том если этот номер больше трех, то я меняю местами все элементы главной и побочной диагоналей матрицы!

А сейчас выберите один из предложенных вариантов:
1:Я сама создам и заполню массив.
2:Вы введёте размер, а заполнение оставите мне.
3:Вы введёте размер и заполните массив.
4:Выход из программы
2
Очень хорошо, в таком случае введите размерность квадратной матрицы:
5
Исходная матрица:
8 8 9 9 2
0 3 2 2 2
7 8 7 7 0
6 5 9 5 2
2 7 7 2 3

Ответ:3
Новая матрица A:
2 8 9 9 8
0 2 2 3 2
7 8 7 7 0
6 5 9 5 2
3 7 7 2 2

Фуууух вроде всё, ну вы это запускайте если что :))!
wfox@Miku:~/C/LabRab_5/Код$

```

Тест 3

Введённые данные: Chose = 3, $n = 3$, $A[i][j] = [1..9]$;

Ожидаемый результат: программа создаст матрицу 3x3 и заполнит её значениями введёнными с клавиатуры (числа от 1 до 9) и, если 3 столбец будет наибольшим, поменяет главную и побочную диагональ;

Полученный результат:

```

Обзор Терминал
wfox@Miku: ~/C/LabRab_5/Код
wfox@Miku:~/C/LabRab_5/Код$ ./Lab5
Привет я Масся, программа которая умеет в матрице натуральных чисел A (размера n*n) определить номер столбца, сумма элементов которого максимальна(беру наибольший номер максимального столбца из равных)!
При том если этот номер больше трех, то я меняю местами все элементы главной и побочной диагоналей матрицы!

А сейчас выберите один из предложенных вариантов:
1:Я сама создам и заполню массив.
2:Вы введёте размер, а заполнение оставите мне.
3:Вы введёте размер и заполните массив.
4:Выход из программы
3
Угу, тогда скорее начнём! Введите размерность квадратной матрицы:
3
а сейчас введите элемент A[1][1]
2
а сейчас введите элемент A[1][2]
1
а сейчас введите элемент A[1][3]
6
а сейчас введите элемент A[2][1]
3
а сейчас введите элемент A[2][2]
5
а сейчас введите элемент A[2][3]
7
а сейчас введите элемент A[3][1]
5
а сейчас введите элемент A[3][2]
8
а сейчас введите элемент A[3][3]
9
Исходная матрица:
2 1 6
3 5 7
5 8 9
Ответ:3
Новая матрица A:
6 1 2
3 5 7
9 8 5
Это всё, если что я буду на вашем электронном носителе :-).
wfox@Miku:~/C/LabRab_5/Код$

```


6 Выводы

По завершению данной лабораторной работы я понял, что такое указатели, динамические и статические массивы, а так же на практике закрепил умения работать с ними.