

Символы, строки и матрицы

Семестр 1

Семинар 4

Символы и строки

```
#include "stdio.h"
#include "stdlib.h"
int main ()
{
    char c = 48, v = '0';
    if (c == v) printf("true\n");
    else printf("false\n");
    printf("%c %d\n", c, v);
    system("pause");
    return 0;
}
```

Символы и строки

```
#include "stdio.h"
#include "stdlib.h"
int main ()
{
    char s[5] = {'H', 'e', 'l', 'l', 'o'},
        d[] = "Hello",
        f[6] = {'H', 'e', 'l', 'l', 'o', '\\0'};
    printf("%s\\n%s\\n%s\\n", s, d, f);
    system("pause");
    return 0;
}
```

Символы и строки

```
#include "stdlib.h"
#include "stdio.h"
#include "string.h"
int main ()
{
    char *a = (char*)malloc(100*sizeof(char)) ;
    scanf("%s", a) ;
    printf("\n%s\n", a) ;
    int n = strlen(a) ;
    a = (char*)realloc(a, n*sizeof(char)) ;
    printf("\n%s\n", a) ;
    free(a) ;
    return 0 ;
}
```

Символы и строки

```
#include "stdlib.h"
#include "stdio.h"
#include "string.h"
int main () {
    char *a = (char*)malloc(100*sizeof(char)), i;
    for (i=-1; (i<99) && (a[i]!='\n');
        a[++i]=getchar());
    a[i] = '\0';
    printf("\n%s\n", a);
    int n = strlen(a);
    a = (char*)realloc(a, n*sizeof(char));
    printf("\n%s\n", a);
    free(a);
    return 0;
}
```

Символы и строки

Функция ввода строки:

0. Объявление массива под «кэш» - статического локального массива символов фиксированного размера *_N*.
1. Объявление указателя на массив символов, который будем возвращать (*char* ret*).
2. Считывание с экрана в кэш посимвольно (*getchar*).
3. Когда закончится кэш – добавить к *ret* количество ячеек памяти *_N* (*realloc*), переписать туда содержимое кэша.
4. Когда закончатся символы – добавить к *ret* количество ячеек памяти *n* (количество символов в кэше), переписать туда содержимое кэша.
5. Вернуть указатель на *ret*.
6. Не забыть в конце программы очистить память!

Двухмерные массивы

```
#include "stdlib.h"
```

```
#include "stdio.h"
```

```
int main () {
```

```
    int m[5][5]={0};
```

```
    for (int i=0; i<5; i++)
```

```
        for (int j=0; j<5; j++)
```

```
            m[i][j] = i*5+j;
```

```
    for (int i=0; i<5; i++)
```

```
        for (int j=0; j<5; j++)
```

```
            printf("%d ", &(m[i][j]));
```

```
printmas(m);
```

```
system("pause");
```

```
return 0;
```

```
}
```

Двухмерные массивы

```
#include "stdlib.h"
#include "stdio.h"
//void printmas(int *m);
//void printmas(int **m);
void printmas(int m[][5]) {
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            printf("%d ",m[i][j]);
}
int main () {
    int m[5][5]={0};
    for (int i=0; i<5; i++)
        for (int j=0; j<5; j++)
            m[i][j] = i*5+j;
    printmas(m);
    system("pause");
    return 0;
}
```


Двухмерные массивы

```
#include "stdlib.h"
#include "stdio.h"
int main () {
    int *m = (int*)malloc(5*5*sizeof(int)) ;

    for (int i=0; i<5; i++)
        for (int j=0; j<5; j++)
            m[i*5+j] = i*5+j; //m[i][j] <- нельзя

    for (int i=0; i<5; i++)
        for (int j=0; j<5; j++)
            printf("%d ",m[i*5+j]) ;

    system("pause") ;
    return 0;
}
```

Двухмерные массивы

```
#include "stdlib.h"
#include "stdio.h"
#define N 10
void printmas(int *m) {
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            printf("%d ",m[i*N+j]);
}
int main () {
    int *m = (int*)malloc(N*N*sizeof(int));
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            m[i*N+j] = i*N+j; //m[i][j] <- нельзя
    printmas(m);
    free(m);
    system("pause"); //только windows
    return 0;
}
```

Двухмерные массивы

```
#include "stdlib.h"
#include "stdio.h"
#define N 5
int main () {
    int **m = (int**)malloc(N*sizeof(int*));
    for (int i=0; i<N; i++)
        m[i] = (int*)malloc(N*sizeof(int));
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            *(*(m+i)+j) = i*N+j;
    *(m[2]+3) = 0;
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            printf("%d ", m[i][j]);
    for (int i=0; i<N; i++)
        free(m[i]);
    free(m)
    system("pause");
    return 0;
}
```

Двухмерные массивы

```
#include "stdlib.h"
#include "stdio.h"
#define N 5
void printmas(int **m) {
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            printf("%d ",m[i][j]);
}
int main () {
    int **m = (int**)malloc(N*sizeof(int*));
    for (int i=0; i<N; i++)
        m[i] = (int*)malloc(N*sizeof(int));
    for (int i=0; i<N; i++)
        for (int j=0; j<N; j++)
            m[i][j] = i*N+j;
    for (int i=0; i<N; i++) free(m[i]);
    free(m)
    system("pause");
    return 0;
}
```

Switch

```
#include "stdlib.h"
#include "stdio.h"
int main () {
    char c = getchar();
    switch (c) {
        case 48:
            printf("zero\n");
            break;
        case '1':
            printf("one\n");
            break;
        case 50:
            printf("two\n");
            break;
        default:
            printf("other\n");
    }
    system("pause");
    return 0;
}
```

Code style

Правила Style checker'а, который будет использоваться на контрольной:

<http://kpm8.mipt.ru:8208/CodeStyle.html>

Задачи

0. Заполнить двумерный массив целых чисел «горизонтальной змейкой», размеры массива вводятся с экрана:

1	2	3	4	5
10	9	8	7	6
11	12	13	14	15

1. Заполнить двумерный массив целых чисел «вертикальной змейкой», размеры массива вводятся с экрана:

1	6	7	12	13
2	5	8	11	14
3	4	9	10	15

2. Умножение матриц $N \times M$ и $M \times L$, содержащих целые числа.
3. Выясните, есть ли одинаковые числа в двумерном массиве произвольных символов.
4. Отсортируйте столбцы массива в порядке возрастания сумм их элементов.
5. Задан двумерный массив из случайных значений. Переставить местами строки так, чтоб главная диагональ содержала значения по убыванию