**Массивы и указатели.**

**Указатели.**

Используются в 4-х случаях:

1. При работе с массивами,
2. При динамическом выделении памяти,
3. При передаче параметров функций,
4. При работе с файлами.

int a=5, \*p, b; // p – указатель (может содержать только // адрес)

p=&a; // занесли в p адрес переменной a.

// & - операция получения адреса

b=\*p; // b стало равно 5, эквивалентно: b=a;

// \* - операция взятия значения по адресу.

int c, \*q;

c=\*&b; // эквивалентно c=b;

q=&\*p; // эквивалентно q=p;

printf(“c=%d, \*q=%d\n”, c, \*q); // c=5, \*q=5

printf(“\*&c=%d, \*q=%d\n”, \*&c, \*&\*q); // \*&c=5, \*&\*q=5

printf(“p=%p, &\*q=%p, \n &c=%p\n”, p, &\*q, &c); //

// p=0x7ffdae1cb2bc, &\*q=0x7ffdae1cb2bc,

// &c=0x7ffdae1cb2c4

**Массивы и указатели.**

**int M[]={1,2,3,4,5}, \*m, N[21]; char S[]=”Just TEXT”, \*s;**

**m=M; // эквивалентно m=&M[0] , \*m==1**

**a=m[4] // эквивалентно a=M[4], a=5**

**a=\*(m+4) // ----“”------“”--------**

**a=\*(M+4) // ----“”------“”--------**

**Общее правило:**

**M[i] ~ \*(M+i) ~ m[i] ~ \*(m+i)**

**m=N; // N=m; N=M; - нельзя, ошибки.**

**m=&N[10]; for(i=-10;i<=10;i++) m[i]=i; // m[0] ~ N[10]**

**m[-10]=-10; m[-9]=-9; … m[0]=0; … m[10]=10;**

**N[0]=-10; N[1]=-9; …**

**С указателями можно использовать отрицательные индексы (с массивами нельзя).**

**m++; // - можно, M++; - нельзя, ошибка.**

**char S1[20]; s=S; i=0; while(S1[i++]=\*s++);**

**Динамическое выделение памяти для массива.**

**printf("Input n="); scanf("%d",&n);// int Massiv[n]; - неверно**

**int \*Massiv=new int[n];**

**for(int i=0;i<n;i++) Massiv[i]=i+1;**

**Динамическое выделение памяти для матрицы.**

**int \*\*A,n,m;int i=0,j=0;printf("Input number of strings: n=");scanf("%d",&n); printf("Input number of columns: m=");scanf("%d",&m);**

**srand(time(0));**

**A= new int\*[n]; for(i=0;i<n;i++) A[i]=new int[m];**

**printf("\nMatrix:\n"); for(i=0;i<n;i++) { for(j=0;j<m;j++) { A[i][j]=rand()%15; printf(" %3d",A[i][j]); } printf("\n"); }// for i**