**目录：**

**1.（一维）将字符数组的元素逐个赋值给整型数组的元素**

**2.输入用户指定行列数的矩阵**

* **定义动态二维数组并读写**

**3.简单数据/数组读取、输出、处理函数**

**4.统计当前的总天数**

**接口参数：年份，月份，日期**

**5.独立字符串处理函数…**

**字符串链接：**

**1.（一维）将字符数组的元素逐个赋值给整型数组的元素**

**接口参数：整型数组名，字符数组名，赋值的长度（元素个数）**

void ArrayToString(int \*num, char \*text, int n)

{

int i;

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

{

\*num = text[i]-48;

}

}

**2.输入用户指定行列数的矩阵**

#define N 100

static int martix[N][N];

int i, j, m, n;

printf("Input m,n:\n");

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

printf("Input matrix:\n");

for (i = 0; i < m; i++) //输入矩阵数据

{

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

{

scanf("%d", &martix[i][j]);

}

}

* **定义动态二维数组并读写**

#include <stdio.h>

#include <stdlib.h>

int \*InputDyArray(const int m, const int n)

{

int i, j;

int \*p = NULL;

p = (int \*)calloc(m\*n, sizeof(int));

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

{

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

{

scanf("%d",(p+i\*n+j));

}

}

return p;

}

1. #include < stdio.h >
2. #include < stdlib.h >
3. **int** \* InputDyArray(**const** **int** m, **const** **int** n)
4. {
5. **int** i, j;
6. **int** \* p;
7. p = (**int** \* ) calloc(m \* n, sizeof(**int**));
8. **for** (i = 0; i < n; i++)
9. {
10. **for** (j = 0; j < n; j++)
11. {
12. scanf("%d", (p + i \* n + j));
13. }
14. }
15. **return** p;
16. }

**3.简单数据/数组读取、输出、处理函数**

（1）.向整型数组中读入整型数据

void ReadData(int a[], int n)

{

int i;

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

{

scanf("%d", &a[i]);

}

}

（2）.由整型数组输出整型数据

void PrintData(int a[], int n)

{

int i;

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

{

printf("%5d", a[i]);

}

printf("\n");

}

（3）.两整数值互换

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

**4.统计当前的总天数**

**接口参数：年份，月份，日期**

**#define YEAR 1990**

**#define DAYS 365**

**int a[] = {0,31,28,31,30,31,30,31,31,30,31,30,31};**

int SumDays(int year,int month,int day)

{

int i = year - YEAR;

int i\_month = 1;

int totalday = day;

if ( (year%4 == 0 && year%100 != 0) || year%400 == 0)

{

a[2] = 29;

}

if (month <= 12 && day <= a[month])

{

while (i\_month < month)

{

totalday += a[i\_month ++];

}

if ( i > 0 )

{

totalday += DAYS\*i + (i-1)/4 + 1;

}

return totalday;

}

else

return -1;

}

**5.独立字符串处理函数…**

**字符串链接：**

char \*MyStrcat(char \*str1, char \*str2)

{

char \*pStr = str1;

while (\*str1 != '\0')

{

str1++;

}

for(; \*str2 != '\0'; str1++,str2++)

{

\*str1 = \*str2;

}

\*str1 = '\0';

return pStr;

}

**6.得到任意整型数的位数：**

int NumBit(long long n)

{

int len;

char text[20];

len = sprintf(text, "%lld", n);

if (n < 0)

len--;

return len;

}