1. 实现n个元素的全排列

void permutation(T list[], int k, int m)

{

if (k == m)

{

copy(list, list + m + 1, ostream\_iterator<T>(cout, "")); //将当前list排序

cout << endl;

}

else{

for (int i = k; i <= m; i++)

{

swap(list[i], list[k]); //将下标为i的元素交换到k位置，类似从list[k:m]中剔除操作

permutation(list, k + 1, m);

swap(list[i], list[k]);

}

}

}

int main(int argc, char\* argv[])

{

char arr[3] = { 'a', 'b', 'c' };

cout << "排序结果如下：" << endl;

permutation(arr, 0, 2);

return 0;

}

1. longlong型数据操作

表示unsigned long long：

printf("num = %llu\n", num);

表示long long int：

printf("num = %lld\n", num);

3.接收

7

5 -3 6 5 -5 -1 6

-6 1 4 -2 0 -2 0

这种格式的方法

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

**for** (**int** i=0; i<n; i++) **scanf**("%d", &a[i]);

**for** (**int** i=0; i<n; i++) **scanf**("%d", &b[i]);

1. C语言字符串分割

#include<string.h>

分解字符串为一组字符串。s为要分解的字符串，delim为分隔符字符串。

strtok()用来将字符串分割成一个个片段。参数s指向欲分割的字符串，参数delim则为分割字符串中包含的所有字符。当strtok()在参数s的字符串中发现参数delim中包涵的分割字符时,则会将该字符改为\0 字符。在第一次调用时，strtok()必需给予参数s字符串，往后的调用则将参数s设置成NULL。每次调用成功则返回指向被分割出片段的指针

s开头开始的一个个被分割的串。当没有被分割的串时则返回NULL。所有delim中包含的字符都会被滤掉，并将被滤掉的地方设为一处分割的节点。

例子：

#include <stdio.h>

#include <conio.h>

int main(void)

{

char input[16]="abc,d,yuwen";

char \*p;

p=strtok(input,",");

while(p!=NULL)

{

printf("%s\n",p);

p=strtok(NULL,",");

}

return 0;

}

输出：

abc

d

yuwen