

Preliminary_Sorting

目的:将无序数组从小到大排序

例如:数组

2	1	3	4
a[0]	a[1]	a[2]	a[3]

- 我们先假设 a[0] 是最大的那个数,则 0 是 maxid.
- 将后面的 a[1] 与 a[maxid] 相比,如果 a[1] > a[maxid] .则 maxid=1
- 再将 a[2] 与 a[maxid] 相比.....
- 得到数组中最大的数 a[maxid]

代码块:

```
/*定义max函数*/
int max(int a[],int length)
{
    int mixid = 0;
    for (int i = 1;i < length;i++){
        if(a[i] > a[maxid]){
            maxid = i;
        }
    }
    return maxid;
}
```

- 得到最大数 a[maxid] 后,需要将a[mixid]放到数组最后的位置 length - 1,同时原来处于最后位置的数放到前面,即交换 a[maxid] 和 a[length - 1]

代码块:

```
int a[]={2,4,1,3};
int length = sizeof(a) / sizeof(a[0]);/*求数组长度的一般方法*/
int i = length - 1

int maxid = max(a,length);
int t = a[maxid];
a[maxid] = a[length - 1];
a[length - 1] = t;/*交换两个值的一般方法*/
```

- 这样就实现了把数组中最大的数放到最后一个位置上,按照这个逻辑重复下去.
- 把第二大的数放到 length-2 的位置上
- 把第三大的数放到 length-3 的位置上
-
- 即 int i = length - 1,每重复一次,就把这个 a[mixid] 放到 i- 的位置上
- 代码块:

```

int main(void)
{
    int a[] = { 2,1,3,4 };
    int length = sizeof(a) / sizeof(a[0]);

    for (int i = length - 1; i > 0; i--) {
        int maxid = max(a, i+1);
        int t = a[maxid];
        a[maxid] = a[i];
        a[i] = t;
    }
    return 0;
}

```

- 最后输出 `a[]`, 即插入代码块:

```

for (int i = 0; i < length; i++) {
    printf("%d ", a[i]);
}

```

完整代码:

```

#include<stdio.h>

int max(int a[], int length);

int main(void)
{
    int a[] = { 2,1,3,4 };
    int length = sizeof(a) / sizeof(a[0]);

    for (int i = length - 1; i > 0; i--) {
        int maxid = max(a, length);
        int t = a[maxid];
        a[maxid] = a[i];
        a[i] = t;
    }
    for (int i = 0; i < length; i++) {
        printf("%d ", a[i]);
    }
    return 0;
}

int max(int a[], int length)
{
    int maxid = 0;
    for (int i = 1; i < length; i++) {
        if (a[i] > a[maxid]) {
            maxid = i;
        }
    }
    return maxid;
}

```

