

## 《面向对象程序设计语言》作业（3.24）

地信班 109092023XXX 许愿

2. 用选择法对 10 个整数排序。

```
#include <iostream>
using namespace std;
int main(){
    int a[10], i, j, t, k; // 初始化
    cout << "请输入 10 个整数: ";
    for (i = 0; i < 10; i++) // 将输入的整数存入数组
        cin >> a[i];
    for (i = 0; i < 9; i++) { // 进行选择排序
        k = i;
        for (j=i+1; j < 10; j++) {
            if (a[j] < a[k]) k = j;
        }
        t = a[k];
        a[k] = a[i];
        a[i] = t;
    }
    cout << "排序后的数组为: "; // 输出排序后的数组
    for (int key : a) {
        cout << key << " ";
    }
    cout << endl;
    return 0;
}
```

```
PS C:\Users\Xuan\Desktop\24-25-2\CPP> & 'c:\
in32-x64\debugAdapters\bin\WindowsDebugLaunc
tdout=Microsoft-MIEngine-Out-zm0f1onf.jal' '
rosoft-MIEngine-Pid-140xzt1j.xfu' '--dbgExe=
请输入10个整数: 3 2 4 6 3 7 1 3 2 2
排序后的数组为: 1 2 2 2 3 3 3 4 6 7
PS C:\Users\Xuan\Desktop\24-25-2\CPP> 
```

3. 有一个已排好序的数组，今输入一个数，要求按原来排序的规律把它插入数组中。

```
#include <iostream>
using namespace std;
int main(){
```

int a[11] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19}, i, j, t, k; // 数组容量要比使用的实际数量大, 空出空间插入

```
cout << "原数组为: "; // 输出原数组
for (int key : a)
    cout << key << " ";
cout << endl;
cout << "请输入一个整数: ";
cin >> t; // 输入整数
int a_length = sizeof(a)/sizeof(a[0]); // 求 a 的长度
bool is_charu = false;
for (i = 0; i < a_length-1; i++) { // 求数组长度, 对数组进行遍历
    if (t < a[i]) { // 若找到应插入 t 的位置
        for (j=a_length-2; j >= i; j--) { // 将 t 位置之后的所有元素进行移动
            a[j + 1] = a[j];
        }
        a[i] = t; // 插入 t
        is_charu = true;
        break;
    }
}
if (!is_charu) { // t 在最末尾插入
    a[a_length-1] = t;
}
cout << "插入后的数组为: ";
for (int key : a) { // 输出插入后的数组
    cout << key << " ";
}
cout << endl;
return 0;
}
```

```
PS C:\Users\Xuan\Desktop\24-25-2\CPP> & 'c:\Users\Xuan\Desktop\24-25-2\CPP\in32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
tdout=Microsoft-MIEngine-Out-bqy4lbqg.snc' '--stderr
rosoft-MIEngine-Pid-3b3bpcfu.we3' '--dbgExe=D:\msys64
原数组为: 1 3 5 7 9 11 13 15 17 19
请输入一个整数: -1
插入后的数组为: -1 1 3 5 7 9 11 13 15 17 19
PS C:\Users\Xuan\Desktop\24-25-2\CPP> & 'c:\Users\Xuan\Desktop\24-25-2\CPP\in32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
tdout=Microsoft-MIEngine-Out-or30o3bb.fye' '--stderr
rosoft-MIEngine-Pid-qb14wmyl.ail' '--dbgExe=D:\msys64
原数组为: 1 3 5 7 9 11 13 15 17 19
请输入一个整数: 8
插入后的数组为: 1 3 5 7 8 9 11 13 15 17 19
PS C:\Users\Xuan\Desktop\24-25-2\CPP> & 'c:\Users\Xuan\Desktop\24-25-2\CPP\in32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
tdout=Microsoft-MIEngine-Out-xrozkw1.ftm' '--stderr
rosoft-MIEngine-Pid-dpmoos15.4ba' '--dbgExe=D:\msys64
原数组为: 1 3 5 7 9 11 13 15 17 19
请输入一个整数: 999
插入后的数组为: 1 3 5 7 9 11 13 15 17 19 999
PS C:\Users\Xuan\Desktop\24-25-2\CPP>
```

5. 将一个数组中的值按逆序重新存放。例如，原来顺序为 8,6,5,4,1, 要求改为 1,4,5,6,8。

```
#include <iostream>
using namespace std;
int main(){
    int a[9] = {8, 6, 7, 2, 1, 9, 3, 2, 1}; // 对数组初始化，这是要进行逆序的数组
    int i,t;
    const int a_length = sizeof(a)/sizeof(a[0]); // 数组长度
    int b[a_length]; // 用于存放逆序后的数组
    // 输出原数组
    cout << "原数组为: ";
    for (int key : a){
        cout << key << " ";
    }
    cout << endl;
    // 逆序将 a 中的数存放到 b 中
    for (i = 0; i < a_length; i++){
        b[a_length - i - 1] = a[i];
    }
    cout << "逆序存放后的数组为: "; // 输出逆序存放后的数组
    for (int key : b){
        cout << key << " ";
    }
    cout << endl;
    // for (i = 0; i < a_length; i++){ // 若需要将 b 中的数存放到 a 中时使用
    //     a[i] = b[i];
    // }
    return 0;
}
```

```
in32-x64\debugAdapters\bin\WindowsDebugLa
原数组为: 8 6 5 4 1
逆序存放后的数组为: 1 4 5 6 8
PS C:\Users\Xuan\Desktop\24-25-2\CPP> & '
in32-x64\debugAdapters\bin\WindowsDebugLa
逆序存放后的数组为: 1 4 5 6 8
PS C:\Users\Xuan\Desktop\24-25-2\CPP> & '
in32-x64\debugAdapters\bin\WindowsDebugLa
tdout=Microsoft-MIEngine-Out-xfhzfzi2.2lu'
in32-x64\debugAdapters\bin\WindowsDebugLa
tdout=Microsoft-MIEngine-Out-xfhzfzi2.2lu'
tdout=Microsoft-MIEngine-Out-xfhzfzi2.2lu'
rosoft-MIEngine-Pid-3ouoihtz.tdc' '--dbgEx
原数组为: 8 6 7 2 1 9 3 2 1
逆序存放后的数组为: 1 2 3 9 1 2 7 6 8
```