

作业讲解

```
#include <iostream>
using namespace std;
int a[100005];
int T[601];
int main(){
    int n,w;
    cin>>n>>w;
    for(int i=1;i<=n;i++){
        cin>>a[i];
        int k=max(1,i*w/100);
        //i个人进行排序
        //所有排序方案理论都行
        //做到i的, 前i-1个人一定有序,
        //想要是的前i个人有序, 只需要把第i个人插入进前i个人里
    }
    //-----
    //计数排序 601*n 100分
    T[a[i]]++;
    int cnt=0,ans=0;
    for(int j=600;j>=0;j--){
        cnt+=T[j]; //加上j分人数
        if(cnt>=k){
            ans=j;
            break;
        }
    }
    cout<<ans<<" ";
    //-----
    //一次合理插入逻辑 时间复杂度: n*n 90分
    for(int j=i;j>1;j--){
        if(a[j]>a[j-1]) swap(a[j],a[j-1]);
        else break;
    }
    //-----
    // //n*n*n 40
    for(int x=1;x<=i-1;x++){
        for(int j=x+1;j<=i;j++){
            if(a[x]<a[j]) swap(a[x],a[j]);
        }
    }
    //-----
    // //70 n*n*logn
    sort(a+1,a+i+1,greater<int>());
    //-----

    cout<<a[k]<<" ";
}

return 0;
}
```

结构体+sort

```
#include <iostream>
#include <algorithm>
using namespace std;

/*
姓名 学号 年龄 成绩(总分)
5
zzq 001 11 305
zby 002 10 303
lzc 003 10 296
whc 004 10 305
www 005 12 300
*/
// int -2e9 <-> 2e9
//unsigned int 无符号整形 0-4e9
//数据类型 int, long long, double, float, char, bool, string, unsigned int
//结构体-用户自己构造的数据类型
/*
struct 类型名{
    元素1;
    元素2;
    .....
};

*/

struct Student{
    char name[30];
    int id;
    int age;
    int score;
};//我们创建了一个自己的数据类型 Student类型
int n;
Student a[100];//100个Student类型的变量

bool mingzi(Student x,Student y){//固定双参数 一个代表排序数组前面的元素 一个代表排序数组
后面的元素
    return x.score>y.score || (x.score==y.score&& x.age<y.age);
}
//成绩由高到低 在成绩相等时 年龄由低到高
//上述两个同时只能满足一个，且我们也只需要满足一个就行
int main(){
    cin>>n;
    for(int i=1;i<=n;i++){
        cin>>a[i].name>>a[i].id>>a[i].age>>a[i].score;
    }

    //sort可以用于所有数组
    sort(a+1,a+n+1,mingzi);//排序规则名 这个规则是一个函数

    cout<<endl;
    for(int i=1;i<=n;i++){
        cout<<a[i].name<<" "<<a[i].score<<endl;
    }
}
```

```
    }  
    return 0;  
}
```

函数

```
#include <iostream>  
using namespace std;  
//函数格式 参考主函数  
  
/*  
//void 无返回值类型  
数据类型 函数名(参数列表){  
  
    return 数据;//函数结束  
    //return返回的和数据类型匹配  
}  
*/  
double f(double x,double y){  
    double ans;  
    if(x>y) ans=x;  
    else ans=y;  
    return ans;  
}  
  
int main(int argc,char** argv){//主函数 一个程序只有一个主函数  
    double x,y,z;  
    cin>>x>>y>>z;  
    cout<<f(f(x,y),z);  
    return 0;  
}
```

函数练习-哥德巴赫猜想

```
#include <iostream>  
using namespace std;  
//哥德巴赫猜想 任意一个大于等8的偶数 等于两个相等的质数的和  
//8-100  
  
bool check(int k){//判断k是否是质数  
    for(int i=2;i<=k-1;i++){  
        if(k%i==0){  
            return false;  
        }  
    }  
    return true;  
}
```

```
void f(int k){//完成对数字k猜想
    for(int i=3;i<=k;i+=2){
        int j=k-i;//所有的奇数对组合 i+j=k
        if(check(i)&&check(j)){//check(i)判断i是否是质数
            cout<<k<<"="<<i<<"+"<<j<<endl;
            return;
        }
    }
}

int main(int argc,char** argv){//主函数 一个程序只有一个主函数
    for(int i=8;i<=100;i+=2){//列举
        f(i);//完成对数字i的哥德巴赫猜想并输出
    }
    return 0;
}
```