排序：

使用C++ 中Algrithm中的 sort函数

Sort(起始地址, 末尾地址+1 , cmp)

Cmp函数的编写

Bool cmp(Student a, Student b)

{

If(a.score != b.score) return a.score > b.score);

Else return strcmp(a.name, b.name)<0;

}

输出排名：

Stu[0].r = 1;

For(int i = 1; i<n;i++)

{

If(stu[i].score == stu[i-1].score){

Stu[i].r = Stu[i-1].r;

}else{

Stu[i].r = i+1;

}

}

Int r = 1;

For(int i =0; i<n;i++)

{

If(i>0 && stu[i].score != stu[i-1].score){

r+=1;

}

// 输出排名，/ stu[i].r = r;

}

1. 德才论

解题思路：

使用结构体 + sort函数

1. The Best Rank

解题思路：

使用结构体 + sort函数

typedef struct{

int id;

int score[4]; //ACME

}Student;

int Rank[1000000][4] = {0}; // 记录每个学生的等级，用 id作为索引

Student student[2005];

bool cmp(Student a, Student b){

return a.score[now]>b.score[now];

} 其中 now为全局变量，通过改变 now来改变排序的成绩科目

1. Phone Bills

解题思路：

1. PAT Ranking

解题思路：

结构体+sort

对每个考场进行排序，把场内排名保存到结构体中，最后进行全排名，直接输出

1. List Sorting

解题思路：全局变量 C来控制 cmp函数

1. The World’s Richest

解题思路：

结构体 + sort函数

对数据进行排序后

可以先将每个年龄的100名存到新的列表中，可以提高查询效率

1. PAT Judge

解题思路：

#include <stdio.h>

#include <string.h>

#include <algorithm>

using namespace std;

typedef struct{

int id;

int score[6];

int sum\_score;

int pcount;

bool doprint;

}Student;

Student student[100000];

bool cmp(Student a, Student b);

int main()

{

int N,K,M;

int p[6];

scanf("%d %d %d", &N,&K,&M);

for(int i = 1; i<=N;i++)

{

student[i].doprint = false;

student[i].id = i;

student[i].pcount = 0;

memset(student[i].score, -1, sizeof(student[i].score));

student[i].sum\_score = 0;

}

for(int i = 1; i<=K;i++)

{

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

}

int temp\_tid,temp\_score,temp\_id;

for(int i =0; i<M;i++)

{

scanf("%d %d %d",&temp\_id ,&temp\_tid, &temp\_score);

if(temp\_score != -1){ //提交通过编译

student[temp\_id].doprint = true;

}

if(temp\_score == -1 && student[temp\_id].score[temp\_tid] == -1){ // 第一次未通过编译，修改成绩为0

temp\_score = 0;

}

if(temp\_score == p[temp\_tid] && student[temp\_id].score[temp\_tid]<p[temp\_tid]){ // 第一次得满分

student[temp\_id].pcount++;

}

if(temp\_score>student[temp\_id].score[temp\_tid]){ //修改score

student[temp\_id].score[temp\_tid] = temp\_score;

}

}

// 计算总分

for(int i = 1; i<= N;i++)

{

for(int j = 1; j<=K;j++)

{

if(student[i].score[j]!=-1){

student[i].sum\_score+=student[i].score[j];

}

}

}

sort(student+1, student+N+1, cmp);

int r = 1;

for(int i = 1; i<=N && student[i].doprint; i++)

{

if(i>1 && student[i].sum\_score != student[i-1].sum\_score){

r = i;

}

printf("%d %05d %d",r,student[i].id,student[i].sum\_score);

for(int j = 1; j<=K;j++)

{

if(student[i].score[j]==-1){

printf(" -");

}else{

printf(" %d", student[i].score[j]);

}

}

printf("\n");

}

return 0;

}

bool cmp(Student a, Student b)

{

if(a.sum\_score!=b.sum\_score) return a.sum\_score > b.sum\_score;

else if(a.pcount != b.pcount) return a.pcount > b.pcount;

else return a.id < b.id;

}

1. List Grades

解题思路：

同The world’s Richest

所有成绩均不同，表明人数做多100

1. Graduate Admission  
   解题思路：
2. Cars on Campus

解题思路：