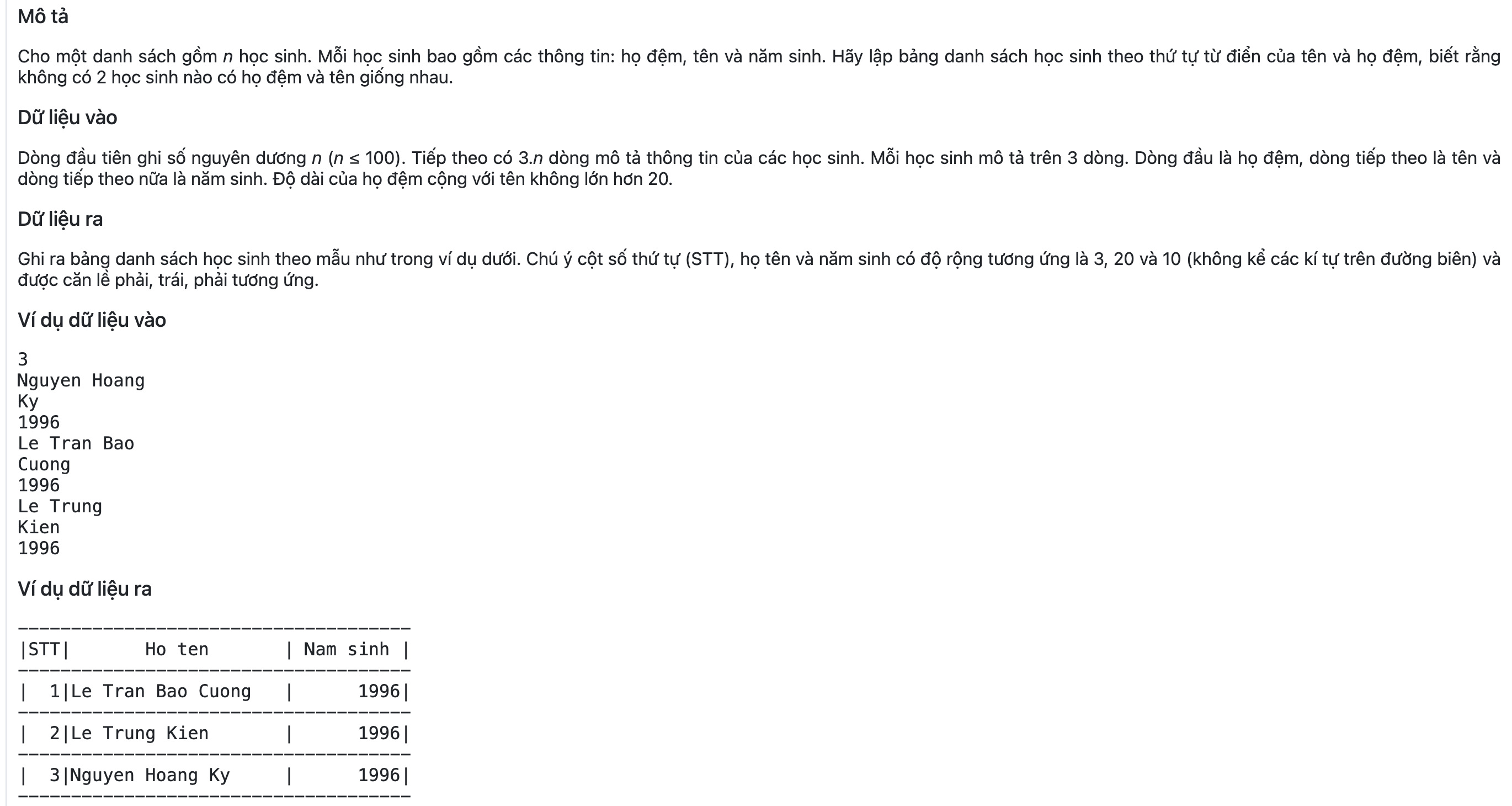


Bài: struct1.cpp #include<bits/stdc++.h>

#include<cstdio>

#include<string>

#define FOR(i,a,b) for(int i=(a);i<=(b);i++)

using namespace std;

struct st{

string hodem, ten;

int nam;

} a[10000];

bool cmp(st a,st b){return a.ten>b.ten;}

int main(){

int n;

cin>>n;

for(int i=1;i<=n;i++){

cin.ignore();

getline(cin,a[i].hodem);

getline(cin,a[i].ten);

cin>>a[i].nam;

}

for(int j=n-1;j>=1;j--)

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

if((a[i].ten>a[i+1].ten) || (a[i].ten==a[i+1].ten && a[i].hodem>a[i+1].hodem))

swap(a[i],a[i+1]);

cout<<"-------------------------------------"<<endl;

cout<<"|STT| Ho ten | Nam sinh |"<<endl;

cout<<"-------------------------------------"<<endl;

for(int i=1;i<=n;i++){

string hoten=a[i].hodem+" "+a[i].ten;

printf("%c%3d%c",'|',i,'|');

printf("%-20s%c",hoten.c\_str(),'|');

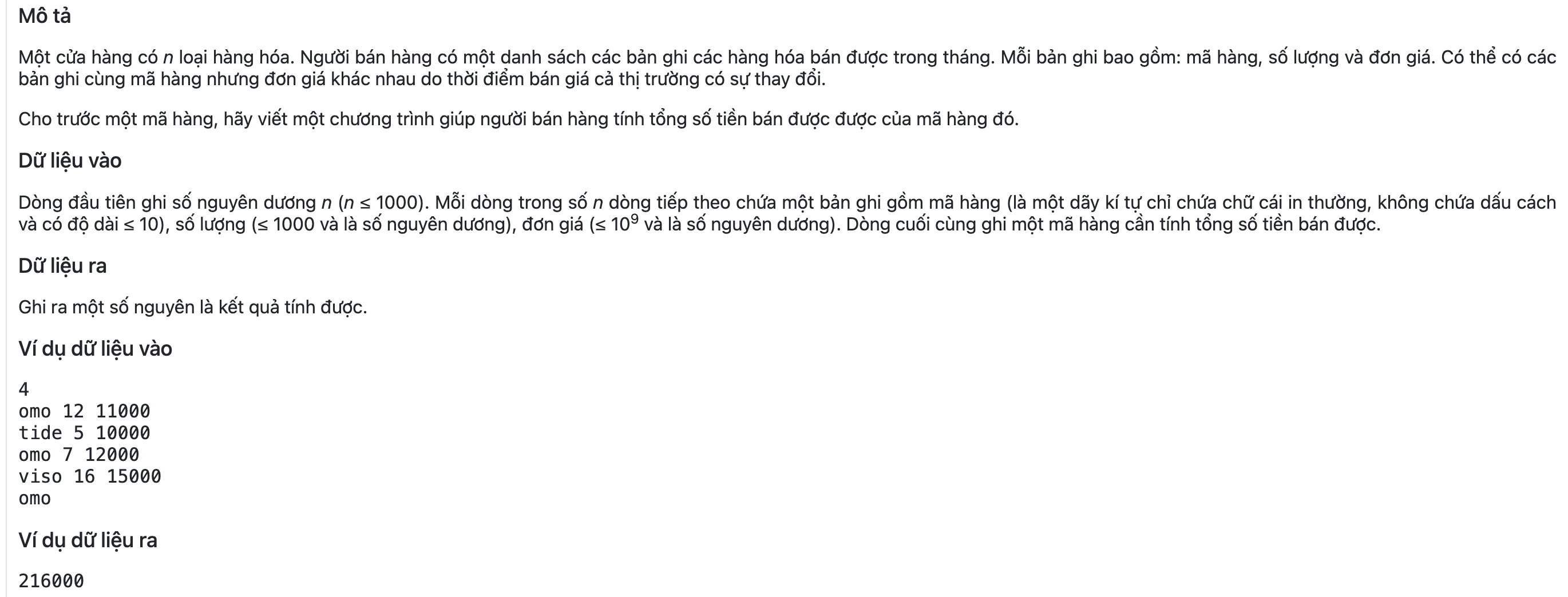
printf("%10d%c\n",a[i].nam,'|');

cout<<"-------------------------------------"<<”\n”;

}

}

Bài: struct2.cpp

 #include<bits/stdc++.h>

#include<cstdio>

#include<string>

#define FOR(i,a,b) for(int i=(a);i<=(b);i++)

using namespace std;

string s;

struct hh{

string ten;

long long sl,dg,tt;

} a[10000];

map <string, long long>ans;

int main(){

int n;

cin>>n;

for(int i=1;i<=n;i++){

cin>>a[i].ten;

cin>>a[i].sl;

cin>>a[i].dg;

a[i].tt=a[i]\*a[i]dg;

ans[a[i].ten]+=a[i].tt;

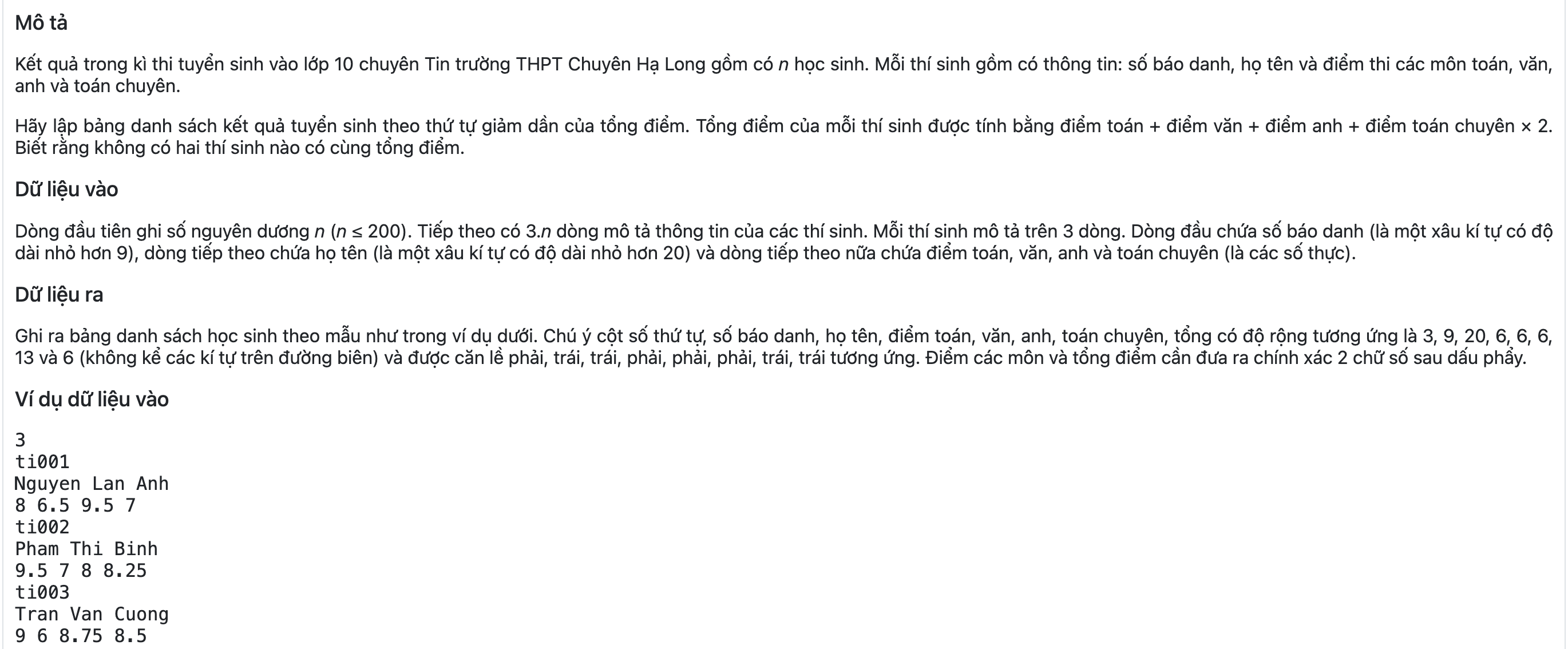
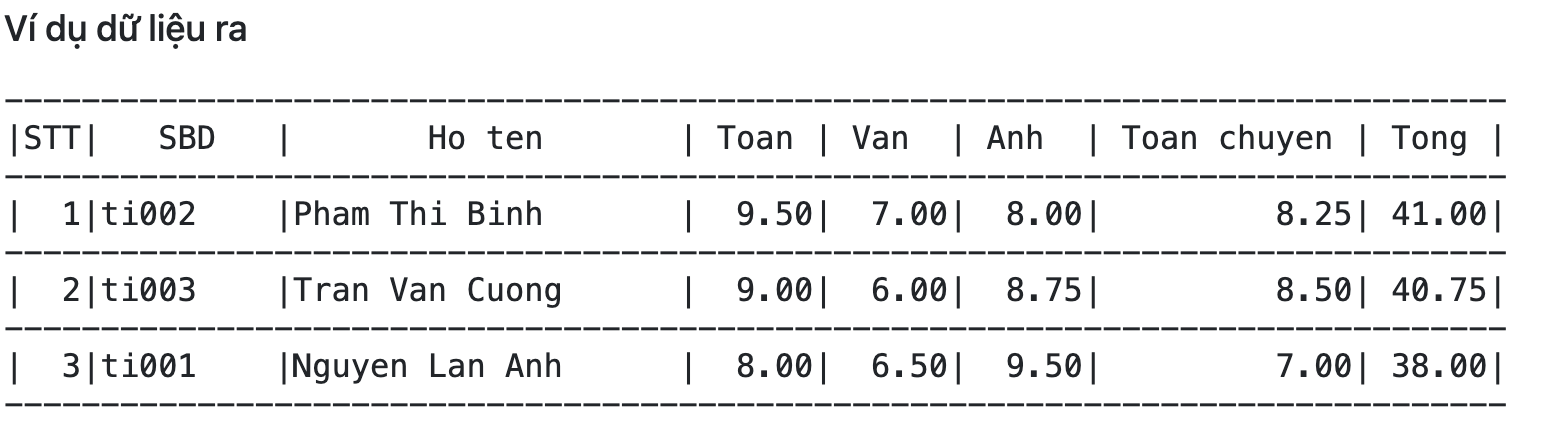
}

cin>>s;

cout<<ans[s]<<endl;

}

Bài: struct3.cpp

  #include<bits/stdc++.h>

using namespace std;

struct hs{

string sbd, ten;

float toan,van,anh,toanchuyen,tong;

} a[10000];

bool cmp(hs a,hs b)

{

return a.tong>b.tong;

}

int main(){

int n;

cin>>n;

cin.ignore();

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

{

getline(cin,a[i].sbd);

getline(cin,a[i].ten);

cin>>a[i].toan>>a[i].van>>a[i].anh>>a[i].toanchuyen;

cin.ignore();

a[i].tong=((a[i].toanchuyen\*2)+a[i].toan+a[i].van+a[i].anh);

}

sort(a+1,a+1+n,cmp);

// In tiêu đề bảng

cout << "------------------------------------------------------------------------------\n";

cout << "|STT| SBD | Ho ten | Toan | Van | Anh | Toan chuyen | Tong |\n";

cout << "------------------------------------------------------------------------------\n";

// In thông tin các thí sinh

for (int i = 1; i <= n; ++i) {

cout << "|";

cout << setw(3) << right << i << "|";

cout << setw(9) << left << a[i].sbd << "|";

cout << setw(20) << left << a[i].ten << "|";

cout << setw(6) << right << fixed << setprecision(2) << a[i].toan << "|";

cout << setw(6) << right << fixed << setprecision(2) << a[i].van << "|";

cout << setw(6) << right << fixed << setprecision(2) << a[i].anh << "|";

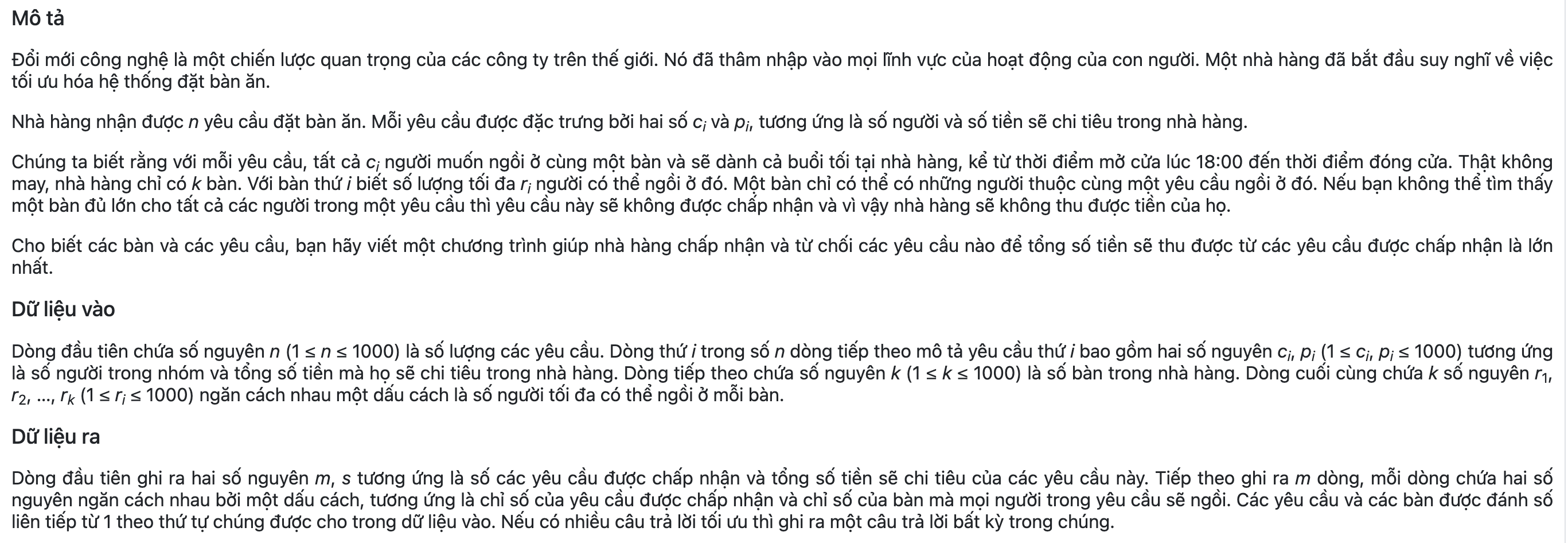
cout << setw(13) << right << fixed << setprecision(2) << a[i].toanchuyen << "|";

cout << setw(6) << right << fixed << setprecision(2) << a[i].tong << "|\n";

}

}

Bài: struct4.cpp

#include<bits/stdc++.h>

using namespace std;

struct yeucau {

int songuoi;

int tien;

int stt; // Số thứ tự của yêu cầu

};

struct ban {

int socho;

int stt; // Số thứ tự của bàn

};

bool cmp1( yeucau a, yeucau b) {

return a.tien > b.tien; // Sắp xếp theo số tiền giảm dần

}

bool cmp2( ban a, ban b) {

return a.socho < b.socho; // Sắp xếp theo sức chứa giảm dần

}

int main() {

int n;

cin >> n;

vector<yeucau> a(n);

for (int i = 0; i < n; ++i) {

cin >> a[i].songuoi >> a[i].tien;

a[i].stt = i + 1; // Chỉ số yêu cầu bắt đầu từ 1

}

int k;

cin >> k;

vector<ban> b(k);

for (int i = 0; i < k; ++i) {

cin >> b[i].socho;

b[i].stt= i + 1; // Chỉ số bàn bắt đầu từ 1

}

// Sắp xếp yêu cầu theo số tiền giảm dần

sort(a.begin(), a.end(), cmp1);

// Sắp xếp bàn theo sức chứa giảm dần

sort(b.begin(), b.end(), cmp2);

vector<bool> usedb(k, false);

vector<pair<int, int>> result; // Kết quả lưu trữ (yêu cầu, bàn)

int totaltien = 0;

for (const auto &req : a) {

for (int j = 0; j < k; ++j) {

if (!usedb[j] && b[j].socho >= req.songuoi) {

// Chấp nhận yêu cầu này với bàn j

result.push\_back({req.stt, b[j].stt});

totaltien += req.tien;

usedb[j] = true;

break;

}

}

}

// In kết quả

cout << result.size() << " " << totaltien << endl;

for (const auto &entry : result) {

cout << entry.first << " " << entry.second << endl;

}

return 0;

} 