Câu 1. Để biết một thí sinh trúng tuyển hay không trong kỳ thi tuyển sinh, chúng ta cần tổ chức dữ liệu gồm các thông tin sau:

Họ tên: họ và tên của thí sinh

Giới tính: nam or nữ

KV: thí sinh thuộc khu vực nào? (1, 2 hay 3)

NH: thí sinh thuộc nhóm nào? (1, 2 hay 3)

TD: tổng điểm 3 môn thi. (Toan, Ly, Hoa)

Hãy nhập một danh sách 10 thí sinh gồm Họ tên,giới tính, KV, NH, TD. Xét xem kết quả thí sinh này đậu hay rớt dựa vào bảng điểm chuẩn

Câu 2. In danh sách đã sắp xếp theo trật tự giảm của TD lên màn hình, gồm các mục Họ tên, KV, NH, TD và Kết quả đậu, rớt.

Câu 3. In danh sách những thí sinh trúng tuyển lên màn hình theo mẫu:

Câu 4: Viết chương trình:

* Ghi toàn bộ dữ liệu danh sách trúng tuyển vào file dau.txt(D:\BaitapC\dau.txt)
* Ghi toàn bộ dữ liệu danh sách không trúng tuyển vào file rot.txt
* Đọc file dau.txt và xuất các bạn là nam.
* Đọc file rot.txt và xuất các bạn là nữ.

#include<stdio.h>

#include<string.h>

struct SinhVien {

char Hoten[100];

char Gioitinh[4];

float KV;

float NH;

float toan,ly,hoa;

float tong3mon;

char ketqua[100];

} sv[100];

void Nhap(int n) {

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

printf("\n Sinh Vien %d\n",i+1);

printf(" Ho Ten: \n");

fflush(stdin);

gets(sv[i].Hoten);

printf(" Gioi tinh: \n");

fflush(stdin);

gets(sv[i].Gioitinh);

do {

printf(" Khu vuc: \n");

fflush(stdin);

scanf("%f",&sv[i].KV);

} while(sv[i].KV>3||sv[i].KV<1);

do {

printf(" Nhom: \n");

fflush(stdin);

scanf("%f",&sv[i].NH);

} while(sv[i].NH>3||sv[i].NH<1);

do {

printf(" toan: \n");

fflush(stdin);

scanf("%f",&sv[i].toan);

} while(sv[i].toan>10||sv[i].toan<0);

do {

printf(" ly: \n");

fflush(stdin);

scanf("%f",&sv[i].ly);

} while(sv[i].ly>10||sv[i].ly<0);

do {

printf(" hoa: \n");

fflush(stdin);

scanf("%f",&sv[i].hoa);

} while(sv[i].hoa>10||sv[i].hoa<0);

printf("-------------------------\n");

}

}

void Xuat( int n) {

printf("\n --------- Thong tin -----\n");

printf("%20s \t %10s \t %6s \t %6s \t %6s\t%20s\n", "Ho Ten", "Gioi Tinh", "Khu Vuc ", "Nhom", "Tong Diem","Ket Qua");

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

printf("%20s \t %10s \t %6.1f \t %6.1f \t %6.2f \t \t %5s\n", sv[i].Hoten, sv[i].Gioitinh, sv[i].KV, sv[i].NH,sv[i].tong3mon,sv[i].ketqua);

}

}

void Tong(SinhVien \*a) {

a->tong3mon=a->toan+a->ly+a->hoa;

}

void KetQua(int n) {

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

if (sv[i].KV==1&&sv[i].NH==1&&sv[i].tong3mon>=18) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==1&&sv[i].NH==1&&sv[i].tong3mon<18) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==1&&sv[i].NH==2&&sv[i].tong3mon>=17.5) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==1&&sv[i].NH==2&&sv[i].tong3mon<17.5) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==1&&sv[i].NH==3&&sv[i].tong3mon>=17) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==1&&sv[i].NH==3&&sv[i].tong3mon<17) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==2&&sv[i].NH==1&&sv[i].tong3mon>=17) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==2&&sv[i].NH==1&&sv[i].tong3mon<17) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==2&&sv[i].NH==2&&sv[i].tong3mon>=16.5) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==2&&sv[i].NH==2&&sv[i].tong3mon<16.5) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==2&&sv[i].NH==3&&sv[i].tong3mon>=16) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==2&&sv[i].NH==3&&sv[i].tong3mon<16) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==3&&sv[i].NH==1&&sv[i].tong3mon>=16) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==3&&sv[i].NH==1&&sv[i].tong3mon<16) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==3&&sv[i].NH==2&&sv[i].tong3mon>=15.5) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==3&&sv[i].NH==2&&sv[i].tong3mon<15.5) {

strcpy(sv[i].ketqua,"R");

}

}

if (sv[i].KV==3&&sv[i].NH==3&&sv[i].tong3mon>=15) {

strcpy(sv[i].ketqua,"D");

} else {

if(sv[i].KV==3&&sv[i].NH==3&&sv[i].tong3mon<15) {

strcpy(sv[i].ketqua,"R");

}

}

}

}

void sapxep(int n) {

for(int i=0; i<n-1; i++) {

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

if(sv[i].tong3mon < sv[j].tong3mon) {

SinhVien tg = sv[i];

sv[i] = sv[j];

sv[j] = tg;

}

}

}

}

void XuatTT ( int n) {

printf("%20s \t %10s \t %6s \t %6s \t %6s\t%20s\n", "Ho Ten", "Gioi Tinh", "Khu Vuc ", "Nhom", "Tong Diem","Ket Qua");

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

if(strcmp(sv[i].ketqua,"D")==0) {

printf("%20s \t %10s \t %6.1f \t %6.1f \t %6.2f \t \t %5s\n", sv[i].Hoten, sv[i].Gioitinh, sv[i].KV, sv[i].NH,sv[i].tong3mon,sv[i].ketqua);

}

}

}

void File( int n) {

char a[100];

FILE \*pt=fopen("D:\\BaitapC\\dau.txt","w");

fprintf(pt,"%20s \t %10s \t %6s \t %6s \t %6s\t%20s\n", "Ho Ten", "Gioi Tinh", "Khu Vuc ", "Nhom", "Tong Diem","Ket Qua");

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

if(strcmp(sv[i].ketqua,"D")==0) {

fprintf(pt,"%20s \t %10s \t %6.1f \t %6.1f \t %6.2f \t \t %5s\n",sv[i].Hoten, sv[i].Gioitinh, sv[i].KV, sv[i].NH,sv[i].tong3mon,sv[i].ketqua);

}

}

fclose(pt);

FILE \*ct=fopen("D:\\BaitapC\\rot.txt","w");

fprintf(ct,"%20s \t %10s \t %6s \t %6s \t %6s\t%20s\n", "Ho Ten", "Gioi Tinh", "Khu Vuc ", "Nhom", "Tong Diem","Ket Qua");

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

if(strcmp(sv[i].ketqua,"R")==0) {

fprintf(ct,"%20s \t %10s \t %6.1f \t %6.1f \t %6.2f \t \t %5s\n",sv[i].Hoten, sv[i].Gioitinh, sv[i].KV, sv[i].NH,sv[i].tong3mon,sv[i].ketqua);

}

}

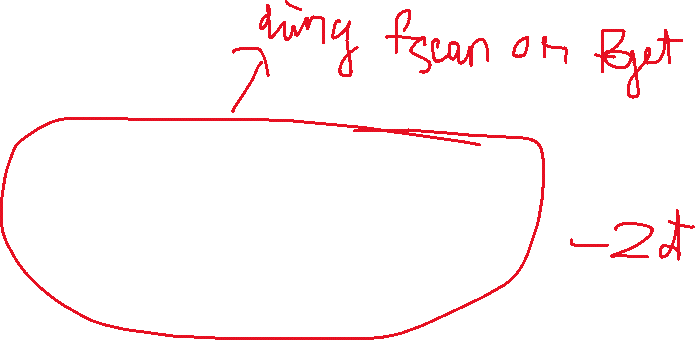
fclose(ct);

}

void FileRaNamDau(int n) {

FILE \*pt =fopen("D:\\BaitapC\\dau.txt","r");

printf("%20s \t %10s \t %6s \t %6s \t %6s\t%20s\n", "Ho Ten", "Gioi Tinh", "Khu Vuc ", "Nhom", "Tong Diem","Ket Qua");



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

if(strcmp(sv[i].Gioitinh,"nam")==0) {

if(strcmp(sv[i].ketqua,"D")==0) {

printf("%20s \t %10s \t %6.1f \t %6.1f \t %6.2f \t \t %5s\n",sv[i].Hoten, sv[i].Gioitinh, sv[i].KV, sv[i].NH,sv[i].tong3mon,sv[i].ketqua);

}

}

}

fclose(pt);

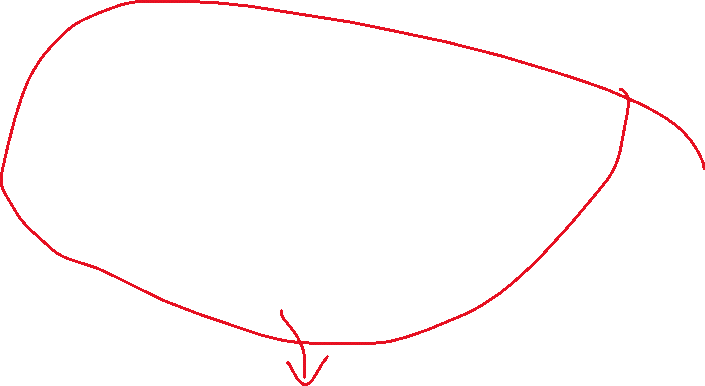
}

void FileRaNuRot(int n) {

FILE \*ct =fopen("D:\\BaitapC\\rot.txt","r");

printf("%20s \t %10s \t %6s \t %6s \t %6s\t%20s\n", "Ho Ten", "Gioi Tinh", "Khu Vuc ", "Nhom", "Tong Diem","Ket Qua");

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



if(strcmp(sv[i].Gioitinh,"nu")==0) {

if(strcmp(sv[i].ketqua,"R")==0) {

printf("%20s \t %10s \t %6.1f \t %6.1f \t %6.2f \t \t %5s\n",sv[i].Hoten, sv[i].Gioitinh, sv[i].KV, sv[i].NH,sv[i].tong3mon,sv[i].ketqua);

}

}

}

fclose(ct);

}



int main () {

/\* CÂU 1\*/

int n;

do {

printf("nhap so sinh vien: ");

scanf("%d",&n);

} while(n<1);

Nhap (n);

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

Tong (&sv[c]);

}

KetQua (n);

Xuat (n);

/\* CÂU 2\*/

printf("\n --------- DANH SACH SAU KHI SAP XEP GIAM -----\n");

sapxep(n);

KetQua (n);

Xuat(n);

/\* CÂU 3\*/

printf("\n --------- DANH SACH THI SINH TRUNG TUYEN -----\n");

XuatTT (n);

/\* CÂU 4\*/

File(n);

printf("\n --------- DANH SACH THI SINH NAM TRONG FILE dau.txt -----\n");

FileRaNamDau(n);

printf("\n --------- DANH SACH THI SINH NU TRONG FILE rot.txt -----\n");

FileRaNuRot(n);

return 0;

}

Câu 2

Hàm void sapxep(int n)

Chart, line chart

Description automatically generated

void KetQua(int n)

Chart

Description automatically generated

Câu 3

Hàm void XuatTT ( int n)

A screenshot of a video game

Description automatically generated

Hàm int main ()

Diagram

Description automatically generated

Calendar

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated