**Bài tập buổi 3 C**

Hình thức nộp: Qua email [dotruc83@gmail.com](mailto:dotruc83@gmail.com)

CÚ PHÁP TIÊU ĐỀ GMAIL: BUOI\_3\_C

TYPE FILE: buoi3C\_họvàtên

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 sau:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Khu vực 1 | Khu vực 2 | Khu vực 3 |
| Nhóm 1 | 18 | 17 | 16 |
| Nhóm 2 | 17.5 | 16.5 | 15.5 |
| Nhóm 3 | 17 | 16 | 15 |

Ví dụ; Thí sinh ở Khu vực 1, Nhóm 2, có Tổng điểm ba môn thi từ 17.5 trở lên thì đậu, ngược lại thì rớt.

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:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TT | Họ và tên | Giới tính | KV | NH | Tổng | Kết quả |
| 1 | Nguyen Van Ban | Nam | 1 | 3 | 17.8 | Đậu |
| 2 | Le Thi Van Thu | Nữ | 3 | 2 | 15 | Rớt |

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.

Chụp toàn bộ kết quả hiển thị trên terminal vào file word

Chụp toàn bộ kết quả 2 file dau.txt, rot.txt

**Bài làm:**

Diagram

Description automatically generated

Timeline

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

#define SoLuong 10

#define DAU\_PATH "D:\\BaitapC\\dau.txt"

#define ROT\_PATH "D:\\BaitapC\\rot.txt"

typedef struct

{

char ten[50];

char gioi[5];

int kv;

int nh;

float td;

char kq[5];

} ThiSinh;

ThiSinh thisinh[SoLuong];

void kiemTraDau(ThiSinh \*ts)

{

float diemChuan[3][3] = {

{18, 17, 16},

{17.5, 16.5, 15.5},

{17, 16, 15}};

if (ts->td >= diemChuan[ts->nh - 1][ts->kv - 1])

strcpy(ts->kq, "Dau");

else

strcpy(ts->kq, "Rot");

}

char \*toLower(char \*s)

{

for (char \*p = s; \*p; p++)

\*p = tolower(\*p);

return s;

}

void nhap()

{

printf("Nhap thong tin cua %d thi sinh\n", SoLuong);

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

{

printf("Thi sinh thu %d\n", i + 1);

printf("\tHo va Ten: ");

fgets(thisinh[i].ten, sizeof(thisinh[i].ten), stdin);

thisinh[i].ten[strlen(thisinh[i].ten) - 1] = '\0';

printf("\tGioi tinh: ");

fgets(thisinh[i].gioi, sizeof(thisinh[i].gioi), stdin);

thisinh[i].gioi[strlen(thisinh[i].gioi) - 1] = '\0';

printf("\tKhu vuc (1-3): ");

scanf("%d", &thisinh[i].kv);

printf("\tNhom (1-3): ");

scanf("%d", &thisinh[i].nh);

printf("\tTong diem: ");

scanf("%f", &thisinh[i].td);

fflush(stdin);

kiemTraDau(&thisinh[i]);

printf("\n");

}

}

void xuatHeader()

{

printf("%-3s | %-20s | %-10s | %-10s | %-10s | %-10s | %-10s\n", "STT", "Ho va Ten", "Gioi tinh", "Khu vuc", "Nhom", "Tong diem", "Ket qua");

for (int i = 0; i < 90; i++)

printf("-");

printf("\n");

}

void xuatHang(int stt, ThiSinh \*ts)

{

printf("%-3d | %-20s | %-10s | %-10d | %-10d | %-10.2f | %-10s\n", stt, ts->ten, ts->gioi, ts->kv, ts->nh, ts->td, ts->kq);

}

void xuatTdGiam()

{

int td[SoLuong][2];

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

{

td[i][0] = thisinh[i].td;

td[i][1] = i;

}

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

{

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

{

if (td[i][0] < td[j][0])

{

int temp = td[i][0];

td[i][0] = td[j][0];

td[j][0] = temp;

temp = td[i][1];

td[i][1] = td[j][1];

td[j][1] = temp;

}

}

}

printf("Danh sach thi sinh sap xep theo tong diem giam dan\n");

xuatHeader();

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

xuatHang(i + 1, &thisinh[td[i][1]]);

}

void xuatDau()

{

int stt = 1;

printf("\nDanh sach thi sinh trung tuyen\n");

xuatHeader();

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

{

if (!strcmp(thisinh[i].kq, "Dau"))

xuatHang(stt++, &thisinh[i]);

}

if (stt == 1)

printf("------ Khong co thi sinh nao trung tuyen ------\n");

}

void luuFile()

{

FILE \*dau = fopen(DAU\_PATH, "w");

FILE \*rot = fopen(ROT\_PATH, "w");

if (dau == NULL || rot == NULL)

{

printf("[ERR] Khong mo duoc file\n");

return;

}

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

{

if (!strcmp(thisinh[i].kq, "Dau"))

{

fprintf(dau, "%s\t", thisinh[i].ten);

fprintf(dau, "%s\t", thisinh[i].gioi);

fprintf(dau, "%d\t", thisinh[i].kv);

fprintf(dau, "%d\t", thisinh[i].nh);

fprintf(dau, "%.2f\t", thisinh[i].td);

fprintf(dau, "%s\n", thisinh[i].kq);

}

else

{

fprintf(rot, "%s\t", thisinh[i].ten);

fprintf(rot, "%s\t", thisinh[i].gioi);

fprintf(rot, "%d\t", thisinh[i].kv);

fprintf(rot, "%d\t", thisinh[i].nh);

fprintf(rot, "%.2f\t", thisinh[i].td);

fprintf(rot, "%s\n", thisinh[i].kq);

}

}

fclose(dau);

fclose(rot);

}

void docFile(char \*fileName, ThiSinh \*ts)

{

char line[500];

char ten[50], gioi[50], kv[50], nh[50], td[50], kq[50];

char \*p[] = {ten, gioi, kv, nh, td, kq};

int i = 0, pi = 0, li = 0;

FILE \*f = fopen(fileName, "r");

if (f == NULL)

{

printf("[ERR] Khong mo duoc file");

printf("%s\n", fileName);

return;

}

while (fgets(line, 500, f) != NULL)

{

i = 0;

pi = 0;

li = 0;

while (1)

{

if (line[li] == '\t')

{

p[pi][i] = '\0';

i = 0;

pi++;

li++;

continue;

}

if (line[li] == '\n')

{

p[pi][i] = '\0';

strcpy(ts->ten, ten);

strcpy(ts->gioi, gioi);

ts->kv = atoi(kv);

ts->nh = atoi(nh);

ts->td = atof(td);

strcpy(ts->kq, kq);

ts++;

break;

}

p[pi][i] = line[li];

i++;

li++;

}

}

fclose(f);

}

void xuatNamDau()

{

int stt = 1;

ThiSinh ts[SoLuong];

docFile(DAU\_PATH, ts);

printf("\nDanh sach thi sinh nam trung tuyen:\n");

xuatHeader();

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

{

if (!strcmp(toLower(ts[i].gioi), "nam"))

xuatHang(stt++, &ts[i]);

}

if (stt == 1)

printf("------ Khong co thi sinh nam trung tuyen ------\n");

}

void xuatNuRot()

{

int stt = 1;

ThiSinh ts[SoLuong];

docFile(ROT\_PATH, ts);

printf("\nDanh sach thi sinh nu rot:\n");

xuatHeader();

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

{

if (!strcmp(toLower(ts[i].gioi), "nu"))

xuatHang(stt++, &ts[i]);

}

if (stt == 1)

printf("------ Khong co thi sinh nu rot ------\n");

}

int main()

{

nhap();

xuatTdGiam();

xuatDau();

luuFile();

xuatNamDau();

xuatNuRot();

return 0;

}

Graphical user interface, text, application

Description automatically generated

A picture containing table

Description automatically generated