Họ và tên: Lê Quang Anh

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.

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated with low confidence

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.

Text

Description automatically generated with low confidence

Diagram

Description automatically generated

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

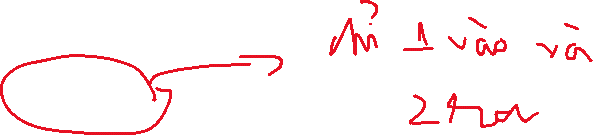
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TT | Họ và tên | Giới tính | KV | NH | Tổng | Kết quả |
| 1 | Nguyen Van Ban | Nam | 1 | 3 | 17.8 | Đậu |
| 2 | Le Thi Van Thu | Nữ | 3 | 2 | 15 | Rớt |
|  |  |  |  |  |  |  |

Text

Description automatically generated

Diagram

Description automatically generated



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.

Text

Description automatically generated

Graphical user interface, application

Description automatically generated

Text

Description automatically generated

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

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

#include<stdio.h>

#include<string.h>

#include<Windows.h>

#define MAX 10

typedef enum {

    rot = 0, dau = 1, khongCo = 3

} kq\_t;

typedef struct HocSinh

{

    char ten[20];

    char gt[4];

    int kv;

    int nh;

    float td;

    kq\_t kq;

} HocSinh\_t;

kq\_t check\_kq(HocSinh\_t hs)

{

    switch (hs.kv)

    {

    case 1:

        switch (hs.nh)

        {

        case 1:

            if (hs.td > 18) return dau;

            else return rot;

            break;

        case 2:

            if (hs.td > 17.5) return dau;

            else return rot;

            break;

        case 3:

            if (hs.td > 17) return dau;

            else return rot;

            break;

        default:

            break;

        }

        break;

    case 2:

        switch (hs.nh)

        {

        case 1:

            if (hs.td > 17) return dau;

            else return rot;

            break;

        case 2:

            if (hs.td > 16.5) return dau;

            else return rot;

            break;

        case 3:

            if (hs.td > 16) return dau;

            else return rot;

            break;

        default:

            break;

        }

        break;

    case 3:

        switch (hs.nh)

        {

        case 1:

            if (hs.td > 16) return dau;

            else return rot;

            break;

        case 2:

            if (hs.td > 15.5) return dau;

            else return rot;

            break;

        case 3:

            if (hs.td > 15) return dau;

            else return rot;

            break;

        default:

            break;

        }

        break;

    default:

        break;

    }

    return khongCo;

}

void chen\_kc(char str1[], int size)

{

    int len = size - strlen(str1);

    for (int i = 0; i < len; i++)

    {

        printf(" ");

    }

}

HocSinh\_t set\_hs(char ten[20], char gt[4], int kv, int nh, float dt)

{

    HocSinh\_t kq;

    strcpy(kq.ten, ten);

    strcpy(kq.gt, gt);

    kq.kv = kv;

    kq.nh = nh;

    kq.td = dt;

    kq.kq = check\_kq(kq);

    return kq;

}

void xuat\_hs(HocSinh\_t hs)

{

    printf("Ho ten: | %s |\r\n", hs.ten);

    printf("Gioi tinh: %s \r\n", hs.gt);

    printf("Khu vuc: %d \r\n", hs.kv);

    printf("Nhom: %d \r\n", hs.nh);

    printf("Diem tb: %.2f \r\n", hs.td);

    printf("Ket qua: %d \r\n", hs.kq);

}

void nhap\_hs(HocSinh\_t \*hs)

{

    while (getchar() != '\n');

    printf("Nhap ho va ten: ");

    gets(hs->ten);

    printf("Nhap gioi tinh: ");

    scanf("%s", &hs->gt);

    while (getchar() != '\n');

    printf("Nhap khu vuc: ");

    scanf("%d", &hs->kv);

    printf("Nhap nhom: ");

    scanf("%d", &hs->nh);

    printf("Nhap tong diem: ");

    scanf("%f", &hs->td);

    hs->kq = check\_kq(\*hs);

}

void nhap\_ds(HocSinh\_t hs[])

{

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

    printf("\t\t\t Nhap danh sach hoc sinh \r\n");

    for (int i = 0; i < MAX; i++)

    {

        printf("\t\tHoc sinh thu %d \r\n", i + 1);

        nhap\_hs(&hs[i]);

    }

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

}

void xuat\_1\_hang(HocSinh\_t hs)

{

    printf(" %s ", hs.ten);

    chen\_kc(hs.ten, 20);

    printf(" |    %s   ", hs.gt);

    chen\_kc(hs.gt, 4);

    printf("| %d  ", hs.kv);

    printf("|  %d   |", hs.nh);

    if (hs.td < 10) {

        printf("  %.2f |", hs.td);

    }

    else

    {

        printf(" %.2f |", hs.td);

    }

    if (hs.kq == dau)

    {

        printf("    dau    |\r\n");

    }

    else

    {

        printf("    rot    |\r\n");

    }

}

void xuat\_bang(HocSinh\_t hs[])

{

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

    printf("| TT | Ho va ten             | Gioi tinh | KV | Nhom | Tong  |  Ket qua  |\r\n");

    for (int i = 0; i < MAX; i++)

    {

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

        printf("| %2d |", i + 1);

        xuat\_1\_hang(hs[i]);

    }

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

}

void swap(HocSinh\_t\* a, HocSinh\_t\* b)

{

    HocSinh\_t temp = \*a;

    \*a = \*b;

    \*b = temp;

}

void sapXep(HocSinh\_t hs[])

{

    for (int i = 0; i < 10 - 1; i++)

    {

        for (int j = i + 1; j < 10; j++)

        {

            if (hs[i].td < hs[j].td) {

                swap(&hs[i], &hs[j]);

            }

        }

    }

}

void dsTrungTuyen(HocSinh\_t hs[])

{

    int cnt = 1;

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

    printf("| TT | Ho va ten             | Gioi tinh | KV | Nhom | Tong  |  Ket qua  |\r\n");

    for (int i = 0; i < 10; i++)

    {

        if (hs[i].kq == dau)

        {

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

            printf("| %2d |", cnt);

            xuat\_1\_hang(hs[i]);

            cnt++;

        }

    }

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

}

void ds\_init(HocSinh\_t a[])

{

    a[0] = set\_hs((char\*)"Nguyen van Yue", (char\*)"nam", 1, 1, 5);

    a[1] = set\_hs((char\*)"Nguyen van Thane", (char\*)"nam", 1, 2, 16);

    a[2] = set\_hs((char\*)"Nguyen van Nak", (char\*)"nam", 1, 3, 9);

    a[3] = set\_hs((char\*)"Nguyen van Zuka", (char\*)"nu", 2, 1, 27);

    a[4] = set\_hs((char\*)"Nguyen van Azenka", (char\*)"nu", 2, 2, 18);

    a[5] = set\_hs((char\*)"Nguyen van violet", (char\*)"nu", 2, 3, 14);

    a[6] = set\_hs((char\*)"Nguyen van Yorn", (char\*)"nam", 3, 2, 25);

    a[7] = set\_hs((char\*)"Nguyen Florentino", (char\*)"nam", 3, 1, 16);

    a[8] = set\_hs((char\*)"Nguyen van Enzo", (char\*)"nam", 3, 3, 9);

    a[9] = set\_hs((char\*)"Nguyen van Preyta", (char\*)"nam", 3, 3, 30);

}

void ds\_Dau\_Rot(HocSinh\_t hs[], HocSinh\_t ds\_dau[], HocSinh\_t ds\_rot[], int \*soDau, int \*soRot)

{

    int s1 = 0, s2 = 0;

    for (int i = 0; i < 10; i++)

    {

        if (hs[i].kq == dau)

        {

            ds\_dau[s1] = hs[i];

            s1++;

        }

        else if (hs[i].kq == rot)

        {

            ds\_rot[s2] = hs[i];

            s2++;

        }

    }

    \*soDau = s1;

    \*soRot = s2;

}

void menu()

{

    printf("\n\t\t Menu \r\n");

    printf("\t1. Nhap danh sach. \r\n");

    printf("\t2. Xuat danh sach sap xep theo thu tu giam dan. \r\n");

    printf("\t3. Xuat danh sach trung tuyen\r\n");

    printf("\t4. Luu file dau va rot. \r\n");

    printf("\t5. Doc file dau va rot. \r\n");

    printf("\t6. Ket thuc. \r\n");

}

void ghifile(HocSinh\_t hs[], const char\* soure, int size)

{

    FILE\* f = fopen(soure, "w");

    for (int i = 0; i < size; i++)

    {

        fprintf(f, "%d, %s, %s, %d, %d, %.2f, %d\n", i, hs[i].ten, hs[i].gt, hs[i].kv, hs[i].nh, hs[i].td, hs[i].kq);

    }

    fclose(f);

}

void docFile(HocSinh\_t hs[], const char\* src)

{

    FILE\* fp = fopen(src, "r");

    char buffer[255];

    int i = 0;

    int d;

    while (fgets(buffer, 255, fp) != NULL)

    {

        // printf("%s", buffer);

        sscanf(buffer, "%d, %[^,], %[^,], %d, %d, %f, %d\n", &d, &hs[i].ten, &hs[i].gt, &hs[i].kv, &hs[i].nh, &hs[i].td, &hs[i].kq);

        printf("%d, %s, %s, %d, %d, %.2f, %d\n", i, hs[i].ten, hs[i].gt, hs[i].kv, hs[i].nh, hs[i].td, hs[i].kq);

        i++;

    }

    fclose(fp);

}

void locData(HocSinh\_t hs[], const char\* gioiTinh)

{

    printf("\n DS Sach Loc theo gioi tinh: %s \r\n", gioiTinh);

    for(int i = 0; i < 10; i++)

    {

        if(strcmp(hs[i].gt, gioiTinh) == 0)

        {

            printf("%d, %s, %s, %d, %d, %.2f, %d\n", i, hs[i].ten, hs[i].gt, hs[i].kv, hs[i].nh, hs[i].td, hs[i].kq);

        }

    }

}

int main()

{

    HocSinh\_t a[10] = { 0 };

    HocSinh\_t ds\_dau[10] = { 0 };

    HocSinh\_t ds\_rot[10] = { 0 };

    HocSinh\_t data\_dau[10] = { 0 };

    HocSinh\_t data\_rot[10] = { 0 };

    int soDau = 0;

    int soRot = 0;

    ds\_init(a);         // dung cai nay de khoi nhap

    int check = 1;

    while (check)

    {

        menu();

        int key;

        printf("\t\t\tNhap chuc nang: ");

        scanf("%d", &key);

        system("cls");

        switch (key)

        {

        case 1:

            nhap\_ds(a);

            printf("\t\t danh sach vua nhap la: \r\n");

            xuat\_bang(a);

            break;

        case 2:

            printf("\r\n\t\t Danh sach sap xep theo thu tu giam dan la: \r\n");

            sapXep(a);

            xuat\_bang(a);

            break;

        case 3:

            printf("\r\n\t\t Danh sach trung tuyen la: \r\n");

            dsTrungTuyen(a);

            break;

        case 4:

            ds\_Dau\_Rot(a, ds\_dau, ds\_rot, &soDau, &soRot);

            ghifile(ds\_dau, "D:\\BaitapC\\dau.txt", soDau);

            ghifile(ds\_rot, "D:\\BaitapC\\rot.txt", soRot);

            printf("OK\nDA LUU\r\n");

            break;

        case 5:

            printf("\t\t Danh sach dau trong file dau.txt \r\n");

            docFile(data\_dau, "D:\\BaitapC\\dau.txt");

            locData(data\_dau, "nam");

            printf("\r\n");

            printf("\t\t Danh sach rot trong file rot.txt \r\n");

            docFile(data\_rot, "D:\\BaitapC\\rot.txt");

            locData(data\_rot, "nu");

            printf("\r\n");

            break;

        case 6:

            check = 0;

            break;

        }

    }

    printf("\t\t\tEnd");

    return 0;

}