2001210004 – Lưu Đức Vinh

Bài tập vê nhà Buổi 3:

Bài tập có hướng dẫn và Bài 1 (thực hành trên lớp):

#define \_CRT\_NONSTDC\_NO\_WARNINGS

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <string.h>

#define MAX 100

struct SV

{

char mssv[11];

char ho[8], hoLot[20], ten[8];

float diemTB;

char xeploai[10];

};

void docFile (FILE \*&f, SV dsSV[], int &n);

void xuat1SV (SV sv);

void xuatdsSV (SV dsSV[], int n);

void ghiFile (FILE \*&f, SV dsSV[], int n);

void showMenu();

void Sapxep\_TangDan\_TheoDTB\_Interchangsort (SV dsSV[], int n);

void Sapxep\_GiamDan\_TheoDTB\_Interchangsort (SV dsSV[], int n);

void Sapxep\_TangDan\_TheoDTB\_Selectionsort (SV dsSV[], int n);

void Sapxep\_GiamDan\_TheoDTB\_Selectionsort (SV dsSV[], int n);

void Sapxep\_TangDan\_TheoDTB\_Quicksort (SV dsSV[], int left, int right);

void Sapxep\_GiamDan\_TheoDTB\_Quicksort (SV dsSV[], int left, int right);

void Tim\_SV\_DiemTB\_Max\_Min\_LinearSearch (SV dsSV[], int n);

void XuatCacSV\_LoaiKhaGioi (SV dsSV[], int n);

void XepLoai (SV dsSV[], int n);

void Dem\_SV\_Ho\_Nguyen (SV dsSV[], int n);

int main ()

{

FILE \*f;

SV dsSV[MAX];

int n, chon;

do

{

showMenu();

printf("Chon chuc nang: "); scanf\_s("%d", &chon);

switch (chon)

{

case 1:

{

docFile(f, dsSV, n);

break;

}

case 2:

{

xuatdsSV(dsSV, n);

break;

}

case 3:

{

ghiFile(f, dsSV, n);

break;

}

case 4:

{

XepLoai(dsSV, n);

break;

}

case 5:

{

int choose;

do

{

printf("1.Tang dan theo Interchangesort\n");

printf("2.Giam dan theo Interchangesort\n");

printf("3.Tang dan theo Selectionsort\n");

printf("4.Giam dan theo Selectionsort\n");

printf("5.Tang dan theo Quicksort\n");

printf("6.Giam dan theo Quicksort\n");

printf("Chon chuc nang: "); scanf\_s("%d", &choose);

switch (choose)

{

case 1:

{

Sapxep\_TangDan\_TheoDTB\_Interchangsort(dsSV, n);

break;

}

case 2:

{

Sapxep\_GiamDan\_TheoDTB\_Interchangsort(dsSV, n);

break;

}

case 3:

{

Sapxep\_TangDan\_TheoDTB\_Selectionsort(dsSV, n);

break;

}

case 4:

{

Sapxep\_GiamDan\_TheoDTB\_Selectionsort(dsSV, n);

break;

}

case 5:

{

Sapxep\_TangDan\_TheoDTB\_Quicksort(dsSV, 0, n - 1);

break;

}

case 6:

{

Sapxep\_GiamDan\_TheoDTB\_Quicksort(dsSV, 0, n - 1);

break;

}

}

}

while (choose != 1 && choose != 2 && choose != 3 && choose != 4 && choose != 5 && choose != 6);

break;

}

case 6:

{

Tim\_SV\_DiemTB\_Max\_Min\_LinearSearch(dsSV, n);

break;

}

case 7:

{

XuatCacSV\_LoaiKhaGioi(dsSV, n);

break;

}

case 8:

{

Dem\_SV\_Ho\_Nguyen(dsSV, n);

break;

}

}

}

while (chon != 0);

return 0;

getch ();

}

void docFile (FILE \*&f, SV dsSV[], int &n)

{

f = fopen("dataStruct.txt", "rt");

if (!f)

{

printf("Loi doc file.");

return;

}

fscanf(f, "%d\n", &n);

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

{

fscanf(f, "%[^#]#%[^#]#%[^#]#%[^#]#%f\n", &dsSV[i].mssv, &dsSV[i].ho, &dsSV[i].hoLot, &dsSV[i].ten, &dsSV[i].diemTB);

}

fclose(f);

}

void ghiFile (FILE \*&f, SV dsSV[], int n)

{

f = fopen("OutputdsSV.txt", "wt");

if (!f)

{

printf("Loi mo file de ghi du lieu.");

return;

}

fprintf(f, "%-5s%-15s%-40s%-10s%-15s\n", "STT", "MSSV", "HO VA TEN SINH VIEN", "DIEM TB", "XEP LOAI");

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

{

fprintf(f, "%-5d", i + 1);

fprintf(f, "%-15s%-10s%-20s%-10s%5.2f%-10s\n", dsSV[i].mssv, dsSV[i].ho, dsSV[i].hoLot, dsSV[i].ten, dsSV[i].diemTB);

}

fclose(f);

}

void xuat1SV (SV sv)

{

printf("%-15s%-10s%-20s%-10s%-10.2f%-10s\n", sv.mssv, sv.ho, sv.hoLot, sv.ten, sv.diemTB, sv.xeploai);

}

void xuatdsSV (SV dsSV[], int n)

{

printf("%-5s%-15s%-40s%-10s%-15s\n", "STT", "MSSV", "HO VA TEN SINH VIEN", "DIEM TB", "XEP LOAI");

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

{

printf("%-5d", i + 1);

xuat1SV(dsSV[i]);

}

}

void XepLoai (SV dsSV[], int n)

{

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

{

if (dsSV[i].diemTB >= 8.0)

{

strcpy(dsSV[i].xeploai, "Gioi");

}

else if (dsSV[i].diemTB >= 6.5)

{

strcpy(dsSV[i].xeploai, "Kha");

}

else if (dsSV[i].diemTB >= 5.0)

{

strcpy(dsSV[i].xeploai, "Trung binh");

}

else

{

strcpy(dsSV[i].xeploai, "Yeu");

}

}

}

void showMenu()

{

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\* MENU \*");

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\* 1. Doc danh sach sinh vien tu file \*");

printf("\n\* 2. Xuat danh sach ra man hinh \*");

printf("\n\* 3. Ghi danh sach sinh vien xuong file \*");

printf("\n\* 4. Xep loai dua theo DTB \*");

printf("\n\* 5. Sap xep danh sach nhan vien tang dan/giam dan theo DTB \*");

printf("\n\* 6. Tim kiem va in ra thong tin SV co diem TB cao nhat/thap nhat \*");

printf("\n\* 7. In ra danh sach cac sinh vien xep loai gioi va kha \*");

printf("\n\* 8. Dem xem co bao nhieu SV ho Nguyen \*");

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

void swap (SV &x, SV &y)

{

SV temp = x;

x = y;

y = temp;

}

void Sapxep\_TangDan\_TheoDTB\_Interchangsort (SV dsSV[], int n)

{

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

{

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

{

if (dsSV[i].diemTB > dsSV[j].diemTB)

{

swap(dsSV[i], dsSV[j]);

}

}

}

}

void Sapxep\_GiamDan\_TheoDTB\_Interchangsort (SV dsSV[], int n)

{

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

{

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

{

if (dsSV[i].diemTB < dsSV[j].diemTB)

{

swap(dsSV[i], dsSV[j]);

}

}

}

}

void Sapxep\_TangDan\_TheoDTB\_Selectionsort (SV dsSV[], int n)

{

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

{

int min = i;

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

{

if (dsSV[min].diemTB > dsSV[j].diemTB)

{

min = j;

}

}

swap(dsSV[min], dsSV[i]);

}

}

void Sapxep\_GiamDan\_TheoDTB\_Selectionsort (SV dsSV[], int n)

{

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

{

int max = i;

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

{

if (dsSV[max].diemTB < dsSV[j].diemTB)

{

max = j;

}

}

swap(dsSV[max], dsSV[i]);

}

}

void Sapxep\_TangDan\_TheoDTB\_Quicksort (SV dsSV[], int left, int right)

{

int i = left, j = right, mid = (left + right) / 2;

SV x = dsSV[mid];

do

{

while (dsSV[i].diemTB < x.diemTB) i++;

while (dsSV[j].diemTB > x.diemTB) j--;

if (i <= j)

{

swap(dsSV[i], dsSV[j]);

i++; j--;

}

}

while (i <= j);

if (left < j) Sapxep\_TangDan\_TheoDTB\_Quicksort(dsSV, left, j);

if (right > i) Sapxep\_TangDan\_TheoDTB\_Quicksort(dsSV, i, right);

}

void Sapxep\_GiamDan\_TheoDTB\_Quicksort (SV dsSV[], int left, int right)

{

int i = left, j = right, mid = (left + right) / 2;

SV x = dsSV[mid];

do

{

while (dsSV[i].diemTB > x.diemTB) i++;

while (dsSV[j].diemTB < x.diemTB) j--;

if (i <= j)

{

swap(dsSV[i], dsSV[j]);

i++; j--;

}

}

while (i <= j);

if (left < j) Sapxep\_GiamDan\_TheoDTB\_Quicksort(dsSV, left, j);

if (right > i) Sapxep\_GiamDan\_TheoDTB\_Quicksort(dsSV, i, right);

}

void Tim\_SV\_DiemTB\_Max\_Min\_LinearSearch (SV dsSV[], int n)

{

float Max = dsSV[0].diemTB;

float Min = dsSV[0].diemTB;

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

{

if (Max < dsSV[i].diemTB)

{

Max = dsSV[i].diemTB;

}

if (Min > dsSV[i].diemTB)

{

Min = dsSV[i].diemTB;

}

}

printf("Sinh vien diem TB cao nhat:\n");

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

{

if (dsSV[i].diemTB == Max)

{

xuat1SV(dsSV[i]);

}

}

printf("Sinh vien diem TB thap nhat:\n");

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

{

if (dsSV[i].diemTB == Min)

{

xuat1SV(dsSV[i]);

}

}

}

void XuatCacSV\_LoaiKhaGioi (SV dsSV[], int n)

{

printf("Danh sach sinh vien xep loai gioi va kha:\n");

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

{

if (stricmp(dsSV[i].xeploai, "Kha") == 0 || stricmp(dsSV[i].xeploai, "Gioi") == 0)

{

xuat1SV(dsSV[i]);

}

}

}

void Dem\_SV\_Ho\_Nguyen (SV dsSV[], int n)

{

int dem = 0;

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

{

if (stricmp(dsSV[i].ho, "Nguyen") == 0)

{

dem++;

}

}

printf("So luong SV ho Nguyen co trong danh sach la: %d\n", dem);

}