Bài 1

#include <stdio.h>

#define MAX 100

struct Student {

char Name[50];

int id;

};

void Nhap(struct Student sv[], int n) {

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

printf("Nhap thong tin sinh vien thu %d\n", i + 1);

printf("Nhap ten cua sinh vien:\n");

fflush(stdin);

gets(sv[i].Name);

printf("Nhap ID cua sinh vien:\n");

scanf("%d", &sv[i].id);

}

}

void XuatDSSV(struct Student sv[], int n) {

printf("Thong tin cua sinh vien\n");

printf("Ten \t ID \n");

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

printf("%s \t %d \n", sv[i].Name, sv[i].id);

}

}

void CapNhapThongTin(struct Student sv[], int n) {

int studentIndex;

printf("Nhap vi tri sinh vien can cap nhat (1 - %d) :", n);

scanf("%d", &studentIndex);

if (studentIndex >= 1 && studentIndex <= n) {

printf("Nhap ten moi:\n");

fflush(stdin);

gets(sv[studentIndex - 1].Name);

printf("Nhap ID moi:\n");

scanf("%d", &sv[studentIndex - 1].id);

} else {

printf("Vi tri sinh vien khong hop le. \n");

}

}

void HienThiThongTin(struct Student sv[], int n) {

printf("Thong tin cua cac sinh vien sau khi cap nhat\n");

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

printf("Thong tin sinh vien thu %d\n", i + 1);

printf("Ten: %s\n", sv[i].Name);

printf("ID: %d\n", sv[i].id);

}

}

int main() {

struct Student sv[MAX];

int n;

printf("Nhap so sinh vien: ");

scanf("%d", &n);

Nhap(sv, n);

XuatDSSV(sv, n);

CapNhapThongTin(sv, n);

HienThiThongTin(sv, n);

return 0;

}

Bài 2

#include <stdio.h>

#include <string.h>

#define MAX 100

struct Student {

char Name[50];

int id;

};

void Nhap(struct Student sv[], int n) {

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

printf("Nhap thong tin sinh vien thu %d\n", i + 1);

printf("Nhap ten cua sinh vien:\n");

scanf("%s", sv[i].Name);

printf("Nhap ID cua sinh vien:\n");

scanf("%d", &sv[i].id);

}

}

void XuatDSSV(struct Student sv[], int n) {

printf("Thong tin cua sinh vien\n");

printf("Ten \t ID \n");

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

printf("%s \t %d \n", sv[i].Name, sv[i].id);

}

}

void SapXepTheoTen(struct Student sv[], int n) {

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

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

if (strcmp(sv[j].Name, sv[j + 1].Name) > 0) {

struct Student temp = sv[j];

sv[j] = sv[j + 1];

sv[j + 1] = temp;

}

}

}

}

void TimKiemTheoTen(struct Student sv[], int n, char ten[]) {

int timThay = 0;

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

if (strcmp(sv[i].Name, ten) == 0) {

printf("Tim thay sinh vien:\n");

printf("Ten: %s\n", sv[i].Name);

printf("ID: %d\n", sv[i].id);

timThay = 1;

break;

}

}

if (!timThay) {

printf("Khong tim thay sinh vien co ten '%s'\n", ten);

}

}

int main() {

struct Student sv[MAX];

int n;

printf("Nhap so sinh vien: ");

scanf("%d", &n);

Nhap(sv, n);

XuatDSSV(sv, n);

SapXepTheoTen(sv, n);

printf("Danh sach sinh vien sau khi sap xep theo ten:\n");

XuatDSSV(sv, n);

char tenTimKiem[50];

printf("Nhap ten sinh vien can tim: ");

scanf("%s", tenTimKiem);

TimKiemTheoTen(sv, n, tenTimKiem);

return 0;

}

Bài 3

#include <stdio.h>

#include <string.h>

#define MAX\_EMPLOYEES 100

struct Employee {

int id;

char name[50];

char position[50];

double salary;

};

void themNhanVien(struct Employee employees[], int \*soLuong) {

if (\*soLuong >= MAX\_EMPLOYEES) {

printf("Đã vượt qua số lượng nhân viên tối đa.\n");

return;

}

struct Employee newEmployee;

printf("Nhập thông tin nhân viên mới:\n");

printf("Nhập mã nhân viên: ");

scanf("%d", &newEmployee.id);

printf("Nhập tên nhân viên: ");

scanf(" %[^\n]s", newEmployee.name);

printf("Nhập chức vụ: ");

scanf(" %[^\n]s", newEmployee.position);

printf("Nhập lương: ");

scanf("%lf", &newEmployee.salary);

employees[\*soLuong] = newEmployee;

(\*soLuong)++;

printf("Nhân viên đã được thêm vào danh sách.\n");

}

void xoaNhanVien(struct Employee employees[], int \*soLuong, int maNhanVien) {

int indexTimThay = -1;

for (int i = 0; i < \*soLuong; i++) {

if (employees[i].id == maNhanVien) {

indexTimThay = i;

break;

}

}

if (indexTimThay == -1) {

printf("Không tìm thấy nhân viên có mã nhân viên %d.\n", maNhanVien);

return;

}

for (int i = indexTimThay; i < \*soLuong - 1; i++) {

employees[i] = employees[i + 1];

}

(\*soLuong)--;

printf("Nhân viên có mã nhân viên %d đã bị xóa khỏi danh sách.\n", maNhanVien);

}

void hienThiNhanVien(struct Employee employees[], int soLuong) {

printf("Danh sách nhân viên:\n");

printf("Mã NV\tTên\tChức vụ\tLương\n");

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

printf("%d\t%s\t%s\t%.2lf\n", employees[i].id, employees[i].name, employees[i].position, employees[i].salary);

}

}

void sapXepNhanVienTheoTen(struct Employee employees[], int soLuong) {

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

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

if (strcmp(employees[j].name, employees[j + 1].name) > 0) {

struct Employee temp = employees[j];

employees[j] = employees[j + 1];

employees[j + 1] = temp;

}

}

}

printf("Danh sách nhân viên sau khi sắp xếp theo tên:\n");

printf("Mã NV\tTên\tChức vụ\tLương\n");

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

printf("%d\t%s\t%s\t%.2lf\n", employees[i].id, employees[i].name, employees[i].position, employees[i].salary);

}

}

int main() {

struct Employee employees[MAX\_EMPLOYEES];

int soLuong = 0;

int luaChon, maNhanVien;

do {

printf("\n---- Quản lý nhân sự ----\n");

printf("1. Thêm nhân viên\n");

printf("2. Xóa nhân viên\n");

printf("3. Hiển thị danh sách nhân viên\n");

printf("4. Sắp xếp nhân viên theo tên\n");

printf("0. Thoát\n");

printf("Nhập lựa chọn của bạn: ");

scanf("%d", &luaChon);

switch (luaChon) {

case 1:

themNhanVien(employees, &soLuong);

break;

case 2:

xoaNhanVien(employees, &soLuong, maNhanVien);

break;

case 3:

hienThiNhanVien(employees, soLuong);

break;

case 4:

sapXepNhanVienTheoTen(employees, soLuong);

break;

case 0:

printf("Chương trình kết thúc.\n");

break;

default:

printf("Lựa chọn không hợp lệ. Vui lòng chọn lại.\n");

break;

}

} while (luaChon != 0);

return 0;

}