Bài thực hành 5:

***Struct***

Bài 1:

#include <stdio.h>

// Dinh nghia cau truc Student

struct Student {

char name[50];

int id;

int age;

float gpa;

};

// Ham nhap thong tin sinh vien

struct Student inputStudent() {

struct Student s;

printf("Nhap ten: ");

fgets(s.name, sizeof(s.name), stdin);

printf("Nhap ma so: ");

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

printf("Nhap tuoi: ");

scanf("%d", &s.age);

printf("Nhap diem trung binh (gpa): ");

scanf("%f", &s.gpa);

getchar(); // Doc ki tu '\n' tu bo nho dem

return s;

}

// Ham hien thi thong tin sinh vien

void displayStudent(struct Student s) {

printf("\nThong tin sinh vien:\n");

printf("Ten: %s", s.name);

printf("Ma so: %d\n", s.id);

printf("Tuoi: %d\n", s.age);

printf("Diem trung binh (gpa): %.2f\n", s.gpa);

}

int main() {

struct Student students[40]; // Mang 40 sinh vien

// Nhap thong tin sinh vien

for (int i = 0; i < 40; i++) {

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

students[i] = inputStudent();

}

// Hien thi thong tin sinh vien

for (int i = 0; i < 40; i++) {

displayStudent(students[i]);

}

return 0;

}

Bài 2:

#include <stdio.h>

#include <string.h>

// Dinh nghia cau truc Student

struct Student {

char name[50];

int id;

int age;

float gpa;

};

// Ham sap xep sinh vien theo ten

void sortStudentsByName(struct Student arr[], int n) {

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

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

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

struct Student temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

// Ham sap xep sinh vien theo diem so (gpa)

void sortStudentsByGrade(struct Student arr[], int n) {

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

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

if (arr[j].gpa > arr[j + 1].gpa) {

struct Student temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

// Ham tim kiem sinh vien theo ten

int searchStudentByName(struct Student arr[], int n, char searchName[]) {

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

if (strcmp(arr[i].name, searchName) == 0) {

return i; // Tra ve vi tri neu tim thay

}

}

return -1; // Tra ve -1 neu khong tim thay

}

// Ham tim kiem sinh vien theo id

int searchStudentById(struct Student arr[], int n, int searchId) {

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

if (arr[i].id == searchId) {

return i; // Tra ve vi tri neu tim thay

}

}

return -1; // Tra ve -1 neu khong tim thay

}

// Ham hien thi thong tin sinh vien

void displayStudent(struct Student s) {

printf("\nThong tin sinh vien:\n");

printf("Ten: %s", s.name);

printf("Ma so: %d\n", s.id);

printf("Tuoi: %d\n", s.age);

printf("Diem: %.2f\n", s.gpa);

}

int main() {

struct Student students[5]; // Mang 5 sinh vien

char searchName[50];

int searchId, result;

// Nhap thong tin sinh vien

for (int i = 0; i < 5; i++) {

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

printf("Nhap ten: ");

fgets(students[i].name, sizeof(students[i].name), stdin);

printf("Nhap ma so: ");

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

printf("Nhap tuoi: ");

scanf("%d", &students[i].age);

printf("Nhap diem (gpa): ");

scanf("%f", &students[i].gpa);

getchar(); // Doc ki tu '\n' tu bo nho dem

}

// Hien thi thong tin sinh vien truoc khi sap xep

printf("\nThong tin sinh vien truoc khi sap xep:\n");

for (int i = 0; i < 5; i++) {

displayStudent(students[i]);

}

// Chon cach sap xep

int choice;

printf("\nChon cach sap xep (1: Theo ten, 2: Theo diem): ");

scanf("%d", &choice);

// Sap xep sinh vien

if (choice == 1) {

sortStudentsByName(students, 5);

} else if (choice == 2) {

sortStudentsByGrade(students, 5);

}

// Hien thi thong tin sinh vien sau khi sap xep

printf("\nThong tin sinh vien sau khi sap xep:\n");

for (int i = 0; i < 5; i++) {

displayStudent(students[i]);

}

// Tim kiem sinh vien

printf("\nNhap ten sinh vien can tim kiem: ");

fgets(searchName, sizeof(searchName), stdin);

searchName[strcspn(searchName, "\n")] = '\0'; // Loai bo ki tu '\n' thua

result = searchStudentByName(students, 5, searchName);

if (result != -1) {

printf("Tim thay sinh vien tai vi tri %d (tinh tu 0)\n", result);

} else {

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

}

// Tim kiem sinh vien theo id

printf("\nNhap ma so sinh vien can tim kiem: ");

scanf("%d", &searchId);

result = searchStudentById(students, 5, searchId);

if (result != -1) {

printf("Tim thay sinh vien tai vi tri %d (tinh tu 0)\n", result);

} else {

printf("Khong tim thay sinh vien co ma so %d\n", searchId);

}

return 0;

}

Bài 3:

#include <stdio.h>

#include <string.h>

// Dinh nghia cau truc Employee

struct Employee {

int employeeId; // Ma nhan vien

char name[50]; // Ten nhan vien

char position[50]; // Chuc vu

float salary; // Luong

};

// Ham nhap thong tin nhan vien

struct Employee inputEmployee() {

struct Employee e;

printf("Nhap ma nhan vien: ");

scanf("%d", &e.employeeId);

getchar(); // Doc ki tu '\n' tu bo nho dem

printf("Nhap ten nhan vien: ");

fgets(e.name, sizeof(e.name), stdin);

e.name[strcspn(e.name, "\n")] = '\0'; // Loai bo ki tu '\n' thua

printf("Nhap chuc vu: ");

fgets(e.position, sizeof(e.position), stdin);

e.position[strcspn(e.position, "\n")] = '\0'; // Loai bo ki tu '\n' thua

printf("Nhap luong: ");

scanf("%f", &e.salary);

return e;

}

// Ham hien thi thong tin nhan vien

void displayEmployee(struct Employee e) {

printf("\nThong tin nhan vien:\n");

printf("Ma nhan vien: %d\n", e.employeeId);

printf("Ten: %s\n", e.name);

printf("Chuc vu: %s\n", e.position);

printf("Luong: %.2f\n", e.salary);

}

// Ham them nhan vien vao mang

void addEmployee(struct Employee arr[], int \*n) {

if (\*n < 10) {

arr[\*n] = inputEmployee();

(\*n)++;

printf("Da them nhan vien thanh cong!\n");

} else {

printf("Danh sach nhan vien da day, khong the them nhan vien moi!\n");

}

}

// Ham xoa nhan vien theo ma nhan vien

int deleteEmployee(struct Employee arr[], int \*n, int employeeId) {

int found = 0;

int newCount = 0; // Bien dem so luong nhan vien moi sau khi xoa

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

if (arr[i].employeeId == employeeId) {

found = 1;

} else {

arr[newCount] = arr[i];

newCount++;

}

}

if (found) {

(\*n)--;

printf("Da xoa nhan vien co ma nhan vien %d!\n", employeeId);

return 1; // Tra ve 1 neu xoa thanh cong

} else {

printf("Khong tim thay nhan vien co ma nhan vien %d!\n", employeeId);

return 0; // Tra ve 0 neu khong tim thay

}

}

// Ham sap xep nhan vien theo ma nhan vien

void sortEmployeesById(struct Employee arr[], int n) {

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

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

if (arr[j].employeeId > arr[j + 1].employeeId) {

struct Employee temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

int main() {

struct Employee employees[10]; // Mang 10 nhan vien

int employeeCount = 0; // So luong nhan vien hien tai

// Them nhan vien

addEmployee(employees, &employeeCount);

// Hien thi thong tin nhan vien

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

displayEmployee(employees[i]);

}

// Them them nhan vien

addEmployee(employees, &employeeCount);

// Hien thi thong tin nhan vien

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

displayEmployee(employees[i]);

}

// Xoa nhan vien theo ma nhan vien

int employeeIdToDelete;

printf("Nhap ma nhan vien can xoa: ");

scanf("%d", &employeeIdToDelete);

deleteEmployee(employees, &employeeCount, employeeIdToDelete);

// Hien thi thong tin nhan vien sau khi xoa

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

displayEmployee(employees[i]);

}

// Sap xep nhan vien theo ma nhan vien

sortEmployeesById(employees, employeeCount);

// Hien thi thong tin nhan vien sau khi sap xep

printf("\nDanh sach nhan vien sau khi sap xep theo ma nhan vien:\n");

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

displayEmployee(employees[i]);

}

return 0;

}

***Array***bài 1:  
#include <stdio.h>

#include <time.h>

void swap(int \*xp, int \*yp) {

int temp = \*xp;

\*xp = \*yp;

\*yp = temp;

}

// Bubble Sort

void bubbleSort(int arr[], int n) {

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

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

if (arr[j] > arr[j+1])

swap(&arr[j], &arr[j+1]);

}

// Selection Sort

void selectionSort(int arr[], int n) {

int i, j, min\_idx;

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

min\_idx = i;

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

if (arr[j] < arr[min\_idx])

min\_idx = j;

swap(&arr[min\_idx], &arr[i]);

}

}

int main() {

int arr[10];

int i;

// Nhap vao

printf("Nhap gia tri cho cac phan tu trong mang:\n");

for (i = 0; i < 10; i++) {

printf("Nhap gia tri cho phan tu thu %d: ", i + 1);

scanf("%d", &arr[i]);

}

// Hien thi mang ban dau

printf("\nMang vua nhap:\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

// Bubble Sort

clock\_t start\_bubble = clock();

bubbleSort(arr, 10);

clock\_t end\_bubble = clock();

// Hien thi mang da sap xep

printf("\nMang sau khi sap xep (Bubble Sort):\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

printf("\nThoi gian thuc thi Bubble Sort: %f giay\n", ((double)end\_bubble - start\_bubble) / CLOCKS\_PER\_SEC);

// Dat lai mang ve trang thai ban dau

printf("\nReset mang ve trang thai ban dau:\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

// Selection Sort

clock\_t start\_selection = clock();

selectionSort(arr, 10);

clock\_t end\_selection = clock();

// Hien thi mang sap xep

printf("\nMang sau khi sap xep (Selection Sort):\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

printf("\nThoi gian thuc thi Selection Sort: %f giay\n", ((double)end\_selection - start\_selection) / CLOCKS\_PER\_SEC);

return 0;

}

Bài 2:

#include <stdio.h>

// Linear Search

int linearSearch(int arr[], int n, int x) {

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

if (arr[i] == x) {

return i; // Tra ve vi tri neu tim thay

}

}

return -1; // Tra ve -1 neu khong tim thay

}

// Bubble Sort (de sap xep mang truoc khi thuc hien Binary Search)

void bubbleSort(int arr[], int n) {

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

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

if (arr[j] > arr[j+1]) {

int temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

// Binary Search

int binarySearch(int arr[], int low, int high, int x) {

while (low <= high) {

int mid = low + (high - low) / 2;

if (arr[mid] == x) {

return mid; // Tra ve vi tri neu tim thay

}

if (arr[mid] < x) {

low = mid + 1;

} else {

high = mid - 1;

}

}

return -1; // Tra ve -1 neu khong tim thay

}

int main() {

int arr[10];

int i, searchValue;

// Input

printf("Nhap gia tri cho cac phan tu trong mang:\n");

for (i = 0; i < 10; i++) {

printf("Nhap gia tri cho phan tu thu %d: ", i + 1);

scanf("%d", &arr[i]);

}

// Display the original array

printf("\nMang vua nhap:\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

// Linear Search

printf("\nNhap gia tri can tim kiem: ");

scanf("%d", &searchValue);

int linearSearchResult = linearSearch(arr, 10, searchValue);

if (linearSearchResult != -1) {

printf("Tim thay tai vi tri %d (tinh tu 0)\n", linearSearchResult);

} else {

printf("Khong tim thay gia tri can tim kiem.\n");

}

// Sorting for Binary Search

bubbleSort(arr, 10);

// Display the sorted array

printf("\nMang sau khi sap xep (Bubble Sort):\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

// Binary Search

printf("\nNhap gia tri can tim kiem (su dung Binary Search): ");

scanf("%d", &searchValue);

int binarySearchResult = binarySearch(arr, 0, 9, searchValue);

if (binarySearchResult != -1) {

printf("Tim thay tai vi tri %d (tinh tu 0)\n", binarySearchResult);

} else {

printf("Khong tim thay gia tri can tim kiem.\n");

}

return 0;

}

Bài 3:

#include <stdio.h>

int main() {

int arr[10];

int i;

float sum = 0;

// Input

printf("Nhap diem cho cac sinh vien trong lop:\n");

for (i = 0; i < 10; i++) {

printf("Nhap diem cho sinh vien thu %d: ", i + 1);

scanf("%d", &arr[i]);

sum += arr[i]; // Tinh tong diem

}

// Display the original array

printf("\nDiem cua cac sinh vien:\n");

for (i = 0; i < 10; i++) {

printf("%d ", arr[i]);

}

// Tinh diem trung binh

float average = sum / 10;

// Display the average

printf("\nDiem trung binh cua lop: %.2f\n", average);

return 0;

}