# Source Code

#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
struct Employee {  
 int id;  
 char name[50];  
 int salary;  
 struct Employee\* next;  
};  
  
void addEmployee(struct Employee\*\* head, int id, char name[], int salary) {  
 struct Employee\* newEmp = (struct Employee\*)malloc(sizeof(struct Employee));  
 newEmp->id = id;  
 strcpy(newEmp->name, name);  
 newEmp->salary = salary;  
 newEmp->next = NULL;  
  
 if (\*head == NULL) {  
 \*head = newEmp;  
 } else {  
 struct Employee\* temp = \*head;  
 while (temp->next != NULL)  
 temp = temp->next;  
 temp->next = newEmp;  
 }  
}  
  
void removeEmployee(struct Employee\*\* head, int id) {  
 struct Employee\* temp = \*head, \*prev = NULL;  
  
 if (temp != NULL && temp->id == id) {  
 \*head = temp->next;  
 free(temp);  
 return;  
 }  
  
 while (temp != NULL && temp->id != id) {  
 prev = temp;  
 temp = temp->next;  
 }  
  
 if (temp == NULL) return;  
  
 prev->next = temp->next;  
 free(temp);  
}  
  
void searchEmployee(struct Employee\* head, int id) {  
 struct Employee\* temp = head;  
 while (temp != NULL) {  
 if (temp->id == id) {  
 printf("Employee Found: %d, %s, %d\n", temp->id, temp->name, temp->salary);  
 return;  
 }  
 temp = temp->next;  
 }  
 printf("Employee with ID %d not found.\n", id);  
}  
  
void displaySortedEmployees(struct Employee\* head) {  
 if (head == NULL) {  
 printf("No employees to display.\n");  
 return;  
 }  
  
 int count = 0;  
 struct Employee\* temp = head;  
 while (temp) {  
 count++;  
 temp = temp->next;  
 }  
  
 struct Employee\* empArray[count];  
 temp = head;  
 for (int i = 0; i < count; i++) {  
 empArray[i] = temp;  
 temp = temp->next;  
 }  
  
 for (int i = 0; i < count - 1; i++) {  
 for (int j = 0; j < count - i - 1; j++) {  
 if (empArray[j]->id > empArray[j + 1]->id) {  
 struct Employee\* temp = empArray[j];  
 empArray[j] = empArray[j + 1];  
 empArray[j + 1] = temp;  
 }  
 }  
 }  
  
 printf("\n--- Employee List (Sorted by ID) ---\n");  
 for (int i = 0; i < count; i++) {  
 printf("ID: %d | Name: %s | Salary: %d\n", empArray[i]->id, empArray[i]->name, empArray[i]->salary);  
 }  
}  
  
int main() {  
 struct Employee\* head = NULL;  
 int choice, id, salary;  
 char name[50];  
  
 do {  
 printf("\n--- Employee Payroll System Menu ---\n");  
 printf("1. Add Employee\n");  
 printf("2. Remove Employee\n");  
 printf("3. Search Employee\n");  
 printf("4. Display All Employees (Sorted by ID)\n");  
 printf("0. Exit\n");  
 printf("Enter choice: ");  
 scanf("%d", &choice);  
  
 switch (choice) {  
 case 1:  
 printf("Enter Employee ID: ");  
 scanf("%d", &id);  
 printf("Enter Name: ");  
 scanf(" %[^  
]", name);  
 printf("Enter Salary: ");  
 scanf("%d", &salary);  
 addEmployee(&head, id, name, salary);  
 break;  
 case 2:  
 printf("Enter Employee ID to remove: ");  
 scanf("%d", &id);  
 removeEmployee(&head, id);  
 break;  
 case 3:  
 printf("Enter Employee ID to search: ");  
 scanf("%d", &id);  
 searchEmployee(head, id);  
 break;  
 case 4:  
 displaySortedEmployees(head);  
 break;  
 case 0:  
 printf("Exiting program...\n");  
 break;  
 default:  
 printf("Invalid option. Try again.\n");  
 }  
  
 } while (choice != 0);  
  
 return 0;  
}