Problem statement

Alumni tracking system

Code

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct Alumni {
int id;
Char name[100];
Char passing year[10];
Struct Alumni *next;
} Alumni;

Alumni *head = NULL;
Alumni * create Alumni (int id, const char*name, const char * passing
                                                        year);
void insert Alumni (int id, const char *name, const char *
                                                    passing year);
Void display Alumni();
void search Alumni();
void update Alumni();
Void delete Alumni();
void flush Input();
void menu();
Alumni * create Alumni (int id, const char * name, const char * passing
                                                        year) {
    Alumni newAlumni = (Alumni) malloc(size of (Alumni));
```

```
if ( !newAlumni)
{
 Printf("Memory Allocation failed \n");
 exit(1);
}
newAlumni->id= id;
Strncpy(newAlumni->name, name, size of (newAlumni->name)-1);
newAlumni->name[size of (newAlumni->name)-1] = '\0';
Strncpy(newAlumni->passing year, passing year,
        Size (newAlumni->passing year[]) -1);
newAlumni->passing year[size of (newAlumni->passing year)-1];
newAlumni->passing year[size of (newAlumni->passing year)-1]='\0';
newAlumni->next =NULL;
return newAlumni;
}
void InsertAlumni (int id, const char *name, const char *passingYear){
Alumni *newAlumni = CreateAlumni (id, name, paring year);
if (!head){
  head=newAlumni;
} else{
  Alumni *temp = head;
while (temp->next) temp=temp->next;
temp->next =newAlumni;
}
Printf("Alumni inserted successful \n");
}
```

```
void displayAlumni()
{
 if(!head)
 {
  Printf("No Alumni records found\n");
 return;
 }
 Printf("Alumni List\n ID\t Name\t\t passing year\n");
 for (Alumni *temp = head; temp != NULL;
     temp = temp->next)
     {
      Printf("%-d\t %-s\t\t %-s\n"
     temp->id, temp->name, temp->passing year);
     }
 }
void flushInput()
{
 int c; while (c=getchar()) != '\n' && c != EOF){}
 }
void searchAlumni()
{
 int choice,id;
 char name[100];
 Alumni *temp = head;
 Printf(" Search by: \n1. ID\n2:Name\n Enter choice: ");
 Scanf("%d", &choice);
 flushInput();
```

```
if (choice == 1)
{
  Printf("Enter Alumni ID: ");
  Scanf("%d", &id);
  flushInput();
while (temp)
{
  if (temp->id == id)
  {
    Printf("Alumni found: \n ID: %d \n Name: %s \n passing year:
                %s\n", temp->id, temp->name,
                temp->passing year);
    return;
  }
  temp = temp->next;
}
Printf("Alumni with ID %d not found \n", id);
}
else if (choice == 2)
{
  Printf("Enter Alumni Name: ");
  fgets(name, size of (name), stdin);
  name [strcspn (name, "\n")]
  = 0;
  while (temp) {
    if (strcmp (temp->name, name) == 0)
    {
      Printf("Alumni found : \n ID: %d \n Name: %. \n passing year:
          %. s\n", temp->id, temp->name, temp->passing year);
      return;
```

```
    }
    temp = temp->next;
  }
  printf("Alumni with name %s not found\n", name);

} else {
    printf("invalid choice\n");
  }
}
void updateAlumni()
{
  int id;
  printf("Enter Alumni ID to update: ");
  scanf("%d", &id);
  flushInput();
  Alumni *temp = head;

else if (choice == 2)
{
  printf("Enter Alumni Name: ");
  fgets

while (temp)
{
  if (temp->id == id) {
    printf("Enter new name");
    fgets(temp->name, sizeof(temp->name), stdin);
    temp->name[strcspn(temp->name, "\n")] = 0;
    printf("Enter new passing year: ");
    scanf("%d", temp->passingyear);
```

```
    flushInput();
    Printf("Alumni record updated succesfully\n");
    return;
  }
  temp = temp->next;
 }
 Printf("Alumni with ID %d not found\n", id);
}
Void deleteAlumni()
{
 int id;
 Printf("Enter Alumni ID to delete:");
 Scanf("%d", &id);
 flushInput();
 Alumni *temp = head, *prev = NULL;
 while (temp) {
    if (temp->id == id)
    {
      if (!prev) head = temp->next;
      else prev->next = temp->next;
    free(temp);
    Printf("Alumni with ID %d deleted Successfully\n", id);
    return;
  }
   prev = temp;
   temp = temp->next;
 }
```

```
Printf("Alumni with ID %d not found \n", id);
}
Void menu(){
  int choice;
  while (1){
  Printf("\n Alumni Tracking System Menu: \n");
  Printf("\n1.Insert Alumni \n 2. display Alumni \n 3. Search Alumni
          \n 4. update Alumni \n 5. Delete Alumni \n 6. Exit \n");
  Printf("Enter your choice:");
  if (Scanf("%d", &choice) != 1){
      Printf("Invalid input \n");
      flush Input();
      Continue;
  }
  flush Input();
  Switch(choice){
  Case 1:{
      int id;
      Char name[100];
      pass year[10];
    Printf("Enter ID:");
    Scanf("%d", &id);
    flush Input();
    Printf("Enter Name:");
    fgets(name, Size of (name), stdin);
    name[strcspn(name, "\n")] = 0;
    Printf("Enter passing year:");
```

```
        scanf("%d", &passing_year);
        flush Input();
        insertAlumni(id, name, passing_year);
        break;
        }
        case 2:
        displayAlumni();
        break;
        case 3:
        searchAlumni();
        break;
        case 4:
        updateAlumni();
        break;
        case 5:
        DeleteAlumni();
        break;
        case 6:
        printf("Existing program\n");
        exit(0);
        default:
        printf("Invalid choice\n");
        }
    }
}
int main()
{
    menu();
    return 0;
}
```