```
Alumhi Tracking System
#include < stdio.h>
# ?ndude < Stdlib. h>
# Include 2 stigng. h>
typedef struct Alumnis
     ant ad;
    char name [so];
    Char department [50];
    int graduation your;
    char occupation [50];
    Starct Alumni *next;
 I Alumni;
 11 Function Prototypes
  Alumni * Greate Alumni (int id, char name [7, char Supt [7,
                           ent year, char occts);
  void Enset Alumni (Alumni ++ head, ent id, chor name []
                     char dept [], Int year, char occij);
   Alumini * Search ByID (Alumni * head, Ent ?d);
  Alumni * searchBy Name (Alumni *head, const char * name);
   Void delete ByID ( Alumni ** head, Int id);
   Void display List (Alumni + head);
       ervorse Display (Alumni + head);
   Void clonelest (Alumni *head, Alumni ** clonettead);
   Void department wiselest (Alumni # head, Const char #dept);
```

```
void Freelist ( Alumni + + head);
Pint mainc) f
    Alumni *head = NULL, * Clone Head = NOIL;
    int chare id, year;
     Char name [50], dept [50], occ [50];
  while (1) §
      Brintf ("In --- Alumhi Tracking system --- In");
      BRNF (" 1. Insert Alumni Ing. search by IDIn3.
                       Search by Name In");
      Printf (4. Delete by IDIns. Display Listing. Revocationly)
      Printfl" 7. Clone Lest Ins. Department - wise Lesting.
               Revesse Lenked Lest In10. Exist In");
      Printy ( " Siter chorce : ");
      scanf ("r.d", & choice);
  switch (choque) 5
       case 1:
        PRINT (" Enter ID: ");
```

void severelest (Alumi *+ head);

scanf ("r.d", 2 id);

get Char();

```
PARNIT (" ENTER Name :");
  feets (name, size of (name), stdin);
  hame [ strcspn (name "In")] = 0;
  PANH (" Enter Department: ");
  fgets ( dept, sixed (dept), sdin);
   dept [strcppn (dept i"In")] = 0;
   Printf ("Enter Graduation your");
   scant (" ".d", 2 years);
    get char();
    PAPULF (" Enter occupation:");
   fgets (occ, sized (occ), stdin);
     occ [stacspn(occ, "In")]=0;
    insert Alumni (2 head, 901, name, dept, year, occ);
     break;
Care d'
    Printf (" Enter 20 to search ;");
    scant ( " %d", 19d);
    Alumni & found ID: search 842D( head, "d);
    if (found 20) 5
       PIRNTF (" Found: "rad 1 %51 7.51 9.0 ( % s/n", foundto >91
             found ID -> name, found ID -> department, found ID
       -> graduation Your, found 10 -> occupation;
```

```
Prentf (" Alumne not food! In");
break;
(asl 3:
     Printf ("Enter Name to search:");
     get Charle);
     fatts (name, six of (name), stdin);
     name [strospn (name, "In")] = 0;
     Alumni + found Name = search By Name (head, name);
     If (found Name) {
           PRINTAC "Founds: rod 18.51 % 51 % of 18.5 ln",
     foundname > id, found Name > name, found Name > department,
    found Name -> graduation year , found Name -> occupation);
     eles?
        Bintof ( " Alumi not found! (n");
       hreak;
    3
 Case 4:
     Right ( " Enter ID to delete; ");
    sconf ( 47d", 29d);
     delete By ID ( lhead, id);
     brack,
```

else 3

```
COURSE 5:
    desplay lest (head);
    break;
 case 6:
     reverse Dinplay (head);
     break,
  Car 7:
       clone List (head, & done Head);
       PRINTF("Cloned Lexts: In");
       display List (clone Head);
       break,
   clase 8:
       Prints (" Enter Department ! ");
        got char();
        fyets (dept, size of (dept), stolen);
        dept [ str cspn (dept, "In") = 0;
        department Wiselist (head, dept);
        break;
     Case 9:
       seversed lest (shead);
       PRINTS (" Linked lest reversed [ In");
       break',
```

```
(ax 10)
       freelPsf (2 head);
       free List (2 clone Head);
       Partity (" Exiting. on In");
        exert (o);
    default :
        printf(" Involed choice! In");
setwin o;
 1/ create new Aleemni node
 Alumnit Create Alumni (int id, char name [], char dept [],
               Port yearty, char occid);
       Alienni + new Alienni = (blumni *) malloc (rige of (Alemni));
       heroAleverini > 9d = 8d;
       stropy (newAlumni > name, name);
      Str cpy ( new Aleumni > department, dept);
      newAlumni -> graduation Year = year;
      Strepy ( new Alumni > occupation, occ);
      new Alumni -> next = NOLL;
       return Rashlumni,
```

```
I Insert at End with Supercale cheak
  Void ansert Alumine (Alumni ++ hood, Put 9d, char name)
            char dept [T, Int year, when occ [T]) f
     of (search By 20 ( * head, id));
       Printf ("ID 7.d already exests! Insertion forled. In", Tal);
       spleven;
   Alumni * new Alumni = create Alumni (id, name, dept, yan, oci);
        if ( * head == NULL) }
            + head = now Alumn?;
          return;
     Alemni + temp = + head;
    while (temp -> next) s
         temp = temp -> next;
        temp>next = navAlunme;
   3
// Search by ID
 Alumni * search842D (Alumni * head, Put 8d) 5
     while (head) f
        if (head - id == id)
           retwen head;
```

```
return NULL;
y
  A search by Name
  Alexani # search By nance (Alexante + head, const char + nance)
    while (head) of
        & ( sto case comp ( head -> name , name) = =0) }
          return bead;
  // Delete Node by ID
   Vogol delete BY ID (Aleemini ## head, int gol) of
        Alumni *temp = * hood, * preve NULL;
    while (temp et temp > 9d; = 9d) f
          prev = temp;
         femp = femp + next;
     if (!femp)s
         Rints (" Record not foundling);
        seturn;
```

head = head -> next;

```
of (prev) f
      Prev -> next = temp -> next;
  alses
      + head = temp - next;
   free (femp);
   Printf(" Alumne with ID " of deleted successfully ( h", "d);
3
11 Display Faul Lest
Vogal display 19st (Alumni * head) 3
    3/ (head) s
       faintef (" No Records (In");
      reterren;
    while (head) 5
      Printe (" yd 1 /s | 2.3 | 2.0 | 2.5 | n"; head -> id,
         head + name, head + department, head + graduation,
      head or occupation);
        head = head > next;
```

```
Il Reversed Derplay using kecursion
 void reverse Draplay (Aleeme + head)
    if (! head ) return;
   reverse Display (head - y next);
  Rint f (" %d 1 %S) 9. Sl 9. d 1 %S In", head -> more id,
   head -> name, head -> deportment, head-> graduatest/out,
 head -> occupation 1;
 Il clone Linked Pert
Void clone List (Alumini & head, Alumi +* clone Hood)?
     * clone Head = NULL'
     Alemon + teil = NOIL;
     while (head)s
      Alumni * hownode = create Alumi (head > id,
   head -> name, head > department, head to graduation your,
  head > occipation);
     if ( * clonel tood = = NULL) ?
          * clone Head = tall = new Noole;
    Jelse {
       fall > next = new Node;
```

```
head = head + next;
11 Department-wise desplay
 Void department hungest ( Alumni , + head, const char + dept)
   Put found =0;
   while ( head) of
     of (str case (mp( head -> department, dept )==0);
          PR wif (" % d 1 % S 1 % S 1 % d 1 % S \n", head -> od,
     head - mame, head -> depart ment, head -> graduation year,
    head -y occupation);
         found = 1;
    head = head -> next;
if (! found) 9
   Printf ("No seconds found for Departments: 1/5/n") dept);
```

fel = Accorde;

```
// Reviews Linked List (actual reversal)
Void revocalist (Alumnia + + thead) of
      Alumni * prev = NOLL, * Cour = + head,
    of mext = NULL;
    while (cure)
       heret = com -> next;
       couls - next = prev;
      Erev = cours;
      Care = next?
   * head = prev;
  Il Free all nodes
Void freelest (Aleerin + + head) q
     Alwans + temp;
 while (# head) }
        temp = * head;
        # head = (* hoad) -> noxt)
        free (temp);
```

```
Programiz
                                                                                                    FRO)
Compile
                            ☐ 5 < Run Output
                                                                                                    Clear
  121 // Create new Alumni node
  122 - Alumni* createAlumni(int id, char name[], char ---- Alumni Tracking System ----
           dept[], int year, char occ[]) { 1. Insert Alumni *newAlumni * (Alumni*)nelloc(sizeof 2. Search by ID
                                                       1. Insert Alumni
  123
              (Alumni));
                                                       3. Search by Nar
           newAlumni->id - id;
  124
                                                       4. Delete by ID
           strcpy(nenAlumni->name, name);
strcpy(nenAlumni->department, dept);
  125
                                                       5. Display List
  126
                                                       6. Reverse Display
  127
           newAlumni->graduationYear = year;
                                                       7. Clone List
  128
           strcpy(nemAlumni ->occupation, occ);
                                                       8. Department-wise List
  129
           newAlumni->next - NULL;
                                                       9. Exit.
          return nenAlumni;
  130
                                                       Enter choice: 1
  131 }
                                                        Enter 10: 0010
  132
                                                        Enter Name: Haripriya
  133 // Insert at End
                                                        Enter Department: AIVL
  134 - void insertAlumni(Alumni **head, int id, char
                                                        Enter Graduation Year: 2028
          name[], char dept[], int year, char occ[]) Enter Occupation: Data Analytics
  135
           Alumni *newAlumni - cresteAlumni(id, name,
                                                        ---- Alumni Tracking System ----
               dept, year, occ);
                                                        1. Insert Alumni
           if ("head == SLLL) {
  136 -
                                                        2. Search by ID
               "head = nemAlumni;
  137
                                                        3. Search by Name
               return:
                                                        4. Delete by ID
  139
                                                        5. Display List
  140
           Alumni "temp = "head;
                                                        6. Reverse Display
           abile (temp->next)
  141
                                                        7. Clone List
  142
              temp + temp- mext;
                                                        8. Department-wise List
  143
           temp--next - newAlumni;
                                                        9. Exit
  144 }
                                                        Enter chaice: 1
  145
                                                        Enter ID: 0011
  146 // Search by ID
                                                        Enter Name: Riya
  147 - Alumni* searchByID(Alumni *head, int id) (
                                                        Enter Department: AIML
  148 -
          while (head) (
                                                         Enter Graduation Year: 2028
              If (head->id == id)
  149
                                                        Enter Occupation: Software developer
  150
                   return head;
               head - head-mext;
  151
                                                         ---- Alumni Tracking System ----
  152
                                                         1. Insert Alumni
          return MULL;
  153
                                                         2. Search by ID
  154
                                                         3. Search by Name
  155
                                                         4. Delete by ID
  156 // Search by Name
                                                         5. Display List
  157 - Alumniº search@yName(Alumni *head, const char
                                                         6. Reverse Display
           'name) {
                                                         7. Clone List
  158 -
           mbile (head) (
                                                         8. Department-wise List
  159
              if (strcasecmp(head->name, name) -- 0)
                                                        9. Exit
  160
                   return head;
                                                         Enter choice: 2
                                                         Enter ID to search: 0010
  161
               head = head->next;
                                                         Found: 10 | Haripriya | AIML | 2028 | Data And
  162
  163
          return MALL:
  164 }
                                                          ---- Alumni Tracking System ----
  165
  166 // Delete Node by ID
                                                          1. Insert Alumni
  167 - void deleteByID(Alumni **head, int id) {
                                                          2. Search by ID
  168
           Alumni *temp - *head, *prev - NULL;
                                                          3. Search by Name
           while (temp && temp->id != id) {
  169 -
                                                          4. Delete by ID
  170
               prev - temp;
                                                          5. Display List
               temp - temp->next;
  171
                                                          6. Reverse Display
  172
                                                          7. Clone List
           if (Itemp) {
  173 -
                                                          8. Department-wise List
  174
               printf("Record not found(\n");
                                                          9. Exit
  175
                                                          Enter choice:
  176
           >
  177
           if (prev)
              prev->next = temp->next;
  178
  179
               "head - temp->next;
  180
  181
           free(temp);
           printf("Alumni with ID %d deleted
  182
               successfully(\n", 1d);
  183 }
  184
  185 // Display Full List
  186 - void displayList(Alumni *head) {
  187 -
          1f (!head) {
               printf("No Records!\n");
  188
  189
               return;
  191 -
           mtile (head) (
               printf("Nd | %s | %s | %d | %s\n".
  192
                   head-rid, head-mane, head
```