```
#include<stdlib.h>
#include<stdio.h>
void create();
void display();
void insertbegin();
void deletebegin();
void deleteend();
void deletepos();

Struct node{ int id,sem;
             char name[20];
             int info;
             Struct node* next; };
Struct node* start=NULL;
int main()
{ int choice;
 while(1){ printf("1Create\n2. Display\n3Inse
     \n 4. Delete from beginning5. Delete from
      the end6. Delete from position\n"7.exit
 printf ("Enter your choice");
 Scanf("%d",&choice);
 Switch(choice)
 { case1: create();
          break;
 Case 2; display(); break;
 Case 3: Insertbegin(); break;
 Case 4: deletebegin(); break;
 Case 5: deleteend(); break;
 Case 6: deletepos(); break;
Case7: exit(0);
        break;
default:
        printf ("Wrong choice\n");
        break;
} } return 0;
  }
void create ()
  { struct node *temp, *ptr;
   temp=(struct node*)malloc(size of(structnod
   if (temp==NULL)
 {
   printf ("Out of Memory Space\n");
     exit (0);
 }
  printf ("Enter input");
 scanf ("%d",&temp->id);
  temp -> next = NULL;
  if (start == NULL)
   { start = temp; }
 else { ptr = start;
       while (ptr->next != NULL)
       { ptr = ptr -> next;
       }
     ptr ->next = temp;
   }
  }
}
```

```
void display()
 {
  Struct node * ptr;
  if (Start = NULL)
  {
    Printf("\nList is empty\n");
     return;
  }
  else
  {
     ptr = Start;
     printf("The list id are \n");
     while (ptr != NULL)
   { printf("%d\t", ptr->d);

     ptr = ptr->next;
   }
  }
}
```

```
void delete_begin()
{
struct node *ptr;
  if (ptr==NULL)
 { printf ("\n List is empty");
    return;
 }
 else
     { ptr = start;
       Start = Start -> next;
      printf (" The deleted item%d \t", ptr->
      free(ptr);
   } }
void delete_end()
{
 Struct node * ptr, temp;
 if (Start == NULL)
  { printf ("List empty);
    exit(0);
  }
 else if (start -> next = NULL)
    {
       ptr= Start;
        Start= NULL;
       printf(" The deleted id is%d", ptr->k
      free(ptr);
  }
 else
    {
```

```
    temp=ptr;
     ptr= ptr->next;
    }
    temp->next = NULL
     free(ptr)
  }
}
 void deletepos()
 {int i,pos;
  struct node* temp,*ptr;
  if (start ==NULL)
  {  printf ("Empty list");
     exit(0) }
 else{
       scanf("%d",&pos)
  if (pos== 0)
    {ptr=start; Start = Start->next;
     printf(~~ptrmd~~"deleted %d", ptr->id);
     free(ptr); }
 else{  ptr= Start;
      for(i= 0;i<pos; i++){temp=ptr;
         ptr= ptr->next;
        if (ptr= NULL)
        { printf ("Position notfound");
             return;} }
    temp->next = ptr->next;
    printf("Deleted element %d",ptr->
    free(ptr); } }  }
```