```
#include < stdio.h >
#include < stdlib.h >
#include<string.h>
struct node {
    int data;
    int priority;
    struct node * next;
};

struct node * start = NULL;
struct node *insert(struct node *start)
{
    int val,pri;
    struct node *ptr,*prev,*cur;
    ptr = (struct node * ) malloc(sizeof(struct node));
    printf("Enter the value and priority:\n");
    scanf("%d%d",&val,&pri);
    ptr->data=val;
    ptr ->priority = pri;
    if (start == NULL || pri < start->priority)
    {
 ptr->next=start;
 start=ptr;
    }
    else
    {
 prev=NULL;
 cur=start;
 while(cur  != NULL && pri >= cur->priority)
 {
    prev=cur;
    cur=cur->next;
 }
```

```
 prev->next=ptr;
 ptr->next=cur;
    }
    return start;
}

struct node *dequeue(struct node *start)
{
    struct node *ptr;

    if (start == NULL)
    {
 printf("\nUnderflow\n");
 exit(0);
    }
    else
    {
 ptr=start;
 printf("\n Deleted item is:%d",ptr->data);
 start=start->next;
 free(ptr);
    }
    return start;
}

void display(struct node *start)
{
    struct node *ptr;
    ptr=start;
    if (start == NULL)
    {
 printf("\nQueue is Empty\n");
    }
```

```
    else
    {
 printf("The queue is \n");
 while (ptr!=NULL)
 {
     printf("\t%d[priority=%d]", ptr-> data,ptr->priority);
     ptr = ptr -> next;
 }

    }
}

int main()
{
    int choice;
    printf("\nImplementation of Queue using Linked List\n");
    while (choice != 4)
    {
 printf("1.Enqueue\n2.Dequeue\n3.Display\n4.Exit\n");
 printf("\nEnter your choice : ");
 scanf("%d", & choice);
 switch (choice)
 {
     case 1: start=insert(start);
  break;
     case 2:
      start=dequeue(start);
  break;
     case 3:
  display(start);
  break;
     case 4:
  exit(0);
```

```
  break;
     default:
  printf("\nWrong Choice\n");
 }
    }
    return 0;
}
```