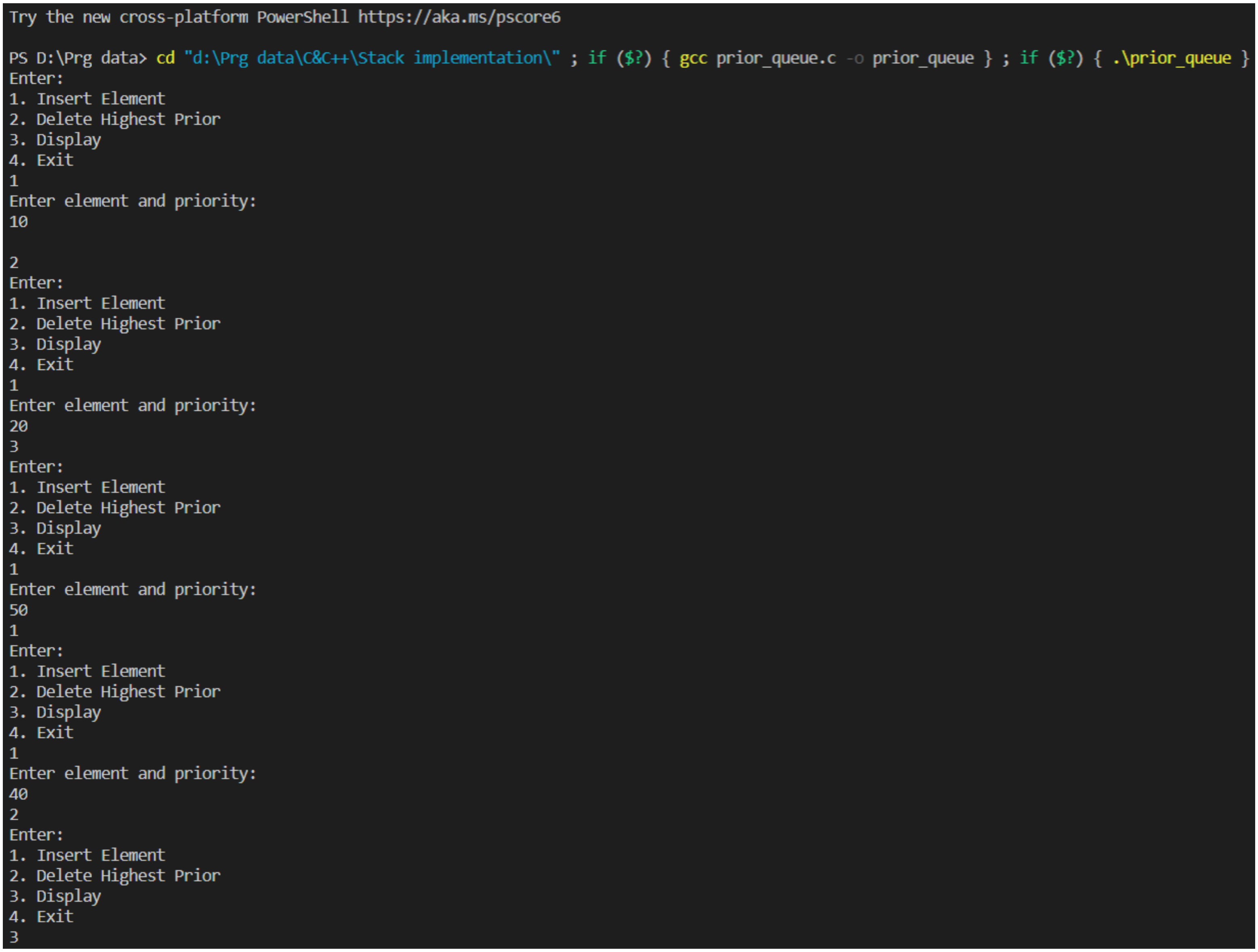
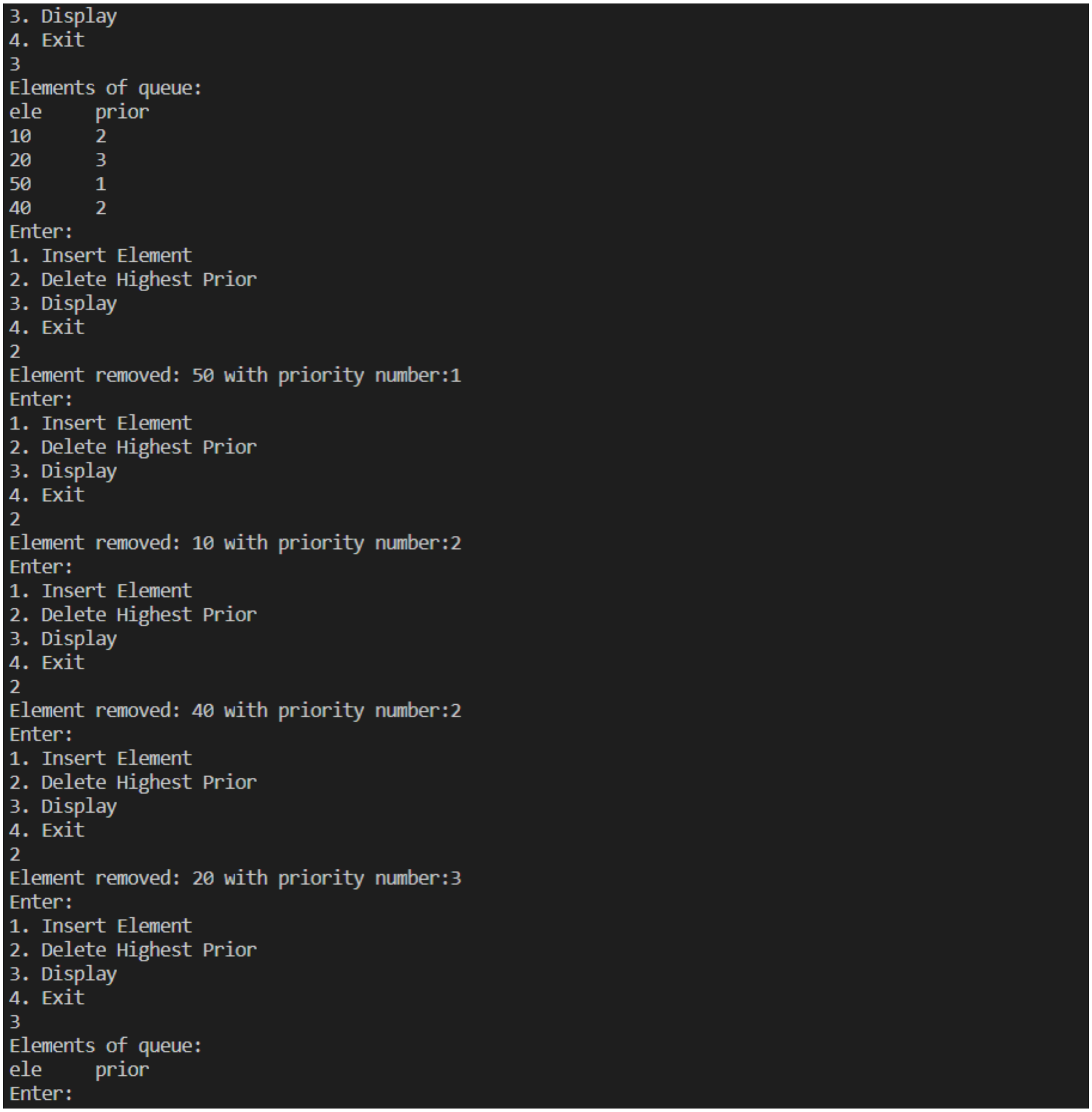
```
}


#include <stdio.h>

#include <stdlib.h>

#include <limits.h>

#define quesize 10


int item, p,  rear=-1, q[quesize][2];


void insrear(){

    if(rear<quesize){

    q[++rear][0]=item;

    q[rear][1]=p;

    }

    else

    printf("Queue overflow\n");

}


void remove_small(){

    int min=INT_MAX;

    int t;

    for(int i=0;i<=rear;i++){

        if(q[i][1]<min){

        min=q[i][1];

        t=i;

        }

    }

    if(min!=INT_MAX){

    printf("Element removed: %d with priority number:%d\n",q[t][0],min);
    q[t][1]=INT_MAX;
    }
    else
    printf("Queue Underflow\n");
}

void display(){
    printf("Elements of queue:\nele\tprior\n");
    for(int i=0;i<=rear;i++){
        if(q[i][1]!=INT_MAX)
        printf("%d\t%d\n",q[i][0],q[i][1]);
    }
}
int main(){
    int choice;
    for(;;){
        printf("Enter:\n1. Insert Element\n2. Delete Highest Prior\n3. Display\n4. Exit\n");
        scanf("%d",&choice);
        switch (choice){
            case 1: printf("Enter element and priority:\n");
            scanf("%d%d", &item,&p);
            insrear();
            break;
            case 2: remove_small();
            break;
            case 3: display();
            break;
```

```
          case 4: exit(0);

          default: printf("Wrong choice\n");

        }

    }

    return 0;

}
```

OUTPUT::



```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Prg data> cd "d:\Prg data\C&C++\Stack implementation\" ; if ($?) { gcc prior_queue.c -o prior_queue } ; if ($?) { .\prior_queue }
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
1
Enter element and priority:
10

2
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
1
Enter element and priority:
20
3
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
1
Enter element and priority:
50
1
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
1
Enter element and priority:
40
2
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
3
```



```
3. Display
4. Exit
3
Elements of queue:
ele     prior
10      2
20      3
50      1
40      2
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
2
Element removed: 50 with priority number:1
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
2
Element removed: 10 with priority number:2
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
2
Element removed: 40 with priority number:2
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
2
Element removed: 20 with priority number:3
Enter:
1. Insert Element
2. Delete Highest Prior
3. Display
4. Exit
3
Elements of queue:
ele     prior
Enter:
```