Lab-6 Ascending Priority Queue

```
#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 %d
                q[t][0], min);
    q[t][1] = INT_MAX;
    }
```

```
else
printf("Queue Underflow \n");
 }
void display() {
 printf("Elements of queue : \n ele \t prior\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 :\n 1. Insert Element \n 2.
                Delete ~~element~~ highest prior \n 3.Displa
               \n 4.Exit\n");
   ~~Swi~~ scanf ("%d", & choice);
      switch (choice) {
       case 1:
            printf("Enter element & priority \n");
          scanf ("%d%d", & item, &p);
          insrear();
          break;
      case 2: remove_small();
          break;
      Case 3: display();
          break;
```

ANKIT IND / 205 X 260 / SINGLE LINE / 53 X 42 / 2 COL / 12-11-19

```
case 4: exit (0);
default: printf(" wrong choice \n"),
}
}
}
return 0;
}
```