LAB-6 Ascending Priority Queue.

```
#include <stdio.h>
# define S5
 int front=0, rear =1;
  int queue [5];

  void insert Rear(){
    int value;
    if (rear==S-1){
        printf ("Queue Overflow\n");
         return;
    }


    int i, j, Small;
    printf ("Enter the value \n");
    Scanf ("%d", & value);
     queue [++rear] = value;
      if (rear >=1) {
            for ( i=1; i<= rear; i++) {
       small = queue [i];
        j=i-1;
          while (j >=0 && queue [j]> small) {
                queue [j+1]= queue [j];
                  j--;
             }
          queue [j+1]= small;
       }
    }
 }
```

```
int deleteFront() {
        if (front > rear) {
        printf ("Queue underflow\n");
          return -1;
  }
    int  value = queue [front ++];
    if (front > rear) {
            front = 0;
             rear = -1;
    }
          return value;
}

void display() {
       int i;
     if (front > rear) {
         printf ("Queue is empty \n");
          return;
      }
   }
   for (i = front; i<= rear; i++) {
        printf ("%d", queue[i]);
   }
     printf ("\n");
}

int main () {
        int  ch, chq, value;
        while (1) {
   printf (" *** Ascanding Priority
                          Queue **");
   printf ("\n1 - Insert to Queue
              \n2 - delete from front
              \n3 - display all Queue
              \n4 - exit \n");
Scanf("%d",&ch);
 Switch( ch) {

   Case1:  InsertRear();
              break;
    Case 2:  value = deleteFront();
             if (value = = -1) { break;}
            else
                  { printf("Deleted item:
                              %d \n", value);
                  }
            break;

     Case 3:
           display();

           break;

       Case 4:
            return 0;
      }
    }
}
```