20) IMPLEMENTING STACKS USING 2 QUEUES

This is the standard sum of the stacks.

CODE OF THE PROGRAM:

```
void QueueStack :: push(int x)
{
   if(q1.size()==0 && q2.size()==0){
      q1.push(x);
   }
   else if(q1.size()==0){
      q2.push(x);
   }
   else if(q2.size()==0){
      q1.push(x);
   }
}

//Function to pop an element from stack using two queues.
int QueueStack :: pop()
{
   if(q1.size()==0 && q2.size()==0){
      return -1;
   }
   else if(q1.size()==0){
      int temp;
      while(q2.size()!=0){
         temp=q2.front();
         q2.pop();
         if(q2.size()!=0){
            q1.push(temp);
         }
      }
      return temp;
   }
   else if(q2.size()==0){
      int temp;
      while(q1.size()!=0){
         temp=q1.front();
         q1.pop();
         if(q1.size()!=0){
            q2.push(temp);
         }
      }
      return temp;
   }
}
```