1. Recall the pseudocode for BFS, and consider the following graph below.

   push start node onto a queue  
   mark start node as visited  
   while queue is not empty:  
     pop node N off queue  
     if N is goal:  
       return true  
     else:  
       for each node O that is child of N:  
         if O is not marked visited:  
           mark node O as visited  
           push O onto queue  
     return false

Find the shortest path starting from B going to E. Record each update (push, pop) to the queue or any returns (true, false) in the table below.

<table>
<thead>
<tr>
<th>Action</th>
<th>Queue Contents</th>
<th>Visited Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>