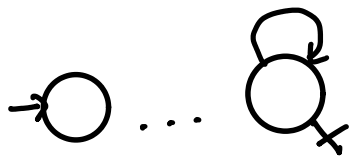
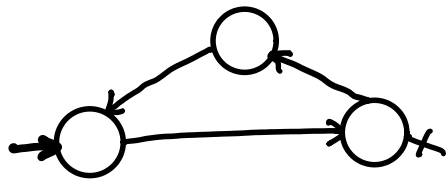


Para que un autómata acepte infinitas palabras debe ser:



Un nodo aceptador que se llame a si mismo.



Que sea un ciclo que contenga un estado final y este estado inicial sea accesible desde el estado final

El algoritmo consiste en dado un nodo final  $q_f$ , ver si existe algun ciclo, es decir si vuelve a visitar el estado  $q_f$

```

procedure BFS( $q_f$ )
  cola := { $q_f$ }
  visitados := { $q_f$ }
  while ( $q_f \neq \emptyset$ )
     $q = \text{cola.pop}()$ 
    for  $p$  adyacente a  $q$ 
      if ( $p == q_f$ )
        return true
      visitado.append( $p$ )
      if (!visitado( $p$ ))
        cola.push( $p$ )
  return false
  
```

```

procedure DFS( $q_f$ )
  visitados =  $\emptyset$ 
  pila =  $q_f$ 
  while pila  $\neq \emptyset$ 
     $q = \text{pila.pop}()$ 
    visitados.append( $q$ )
    para cada  $p$  adyacente a  $q$ 
      si  $p == q$ 
        return true
      si  $p \notin \text{visitados}$ 
        pila.push( $p$ )
  return false
  
```

Estos dos algoritmos solo te dicen si existe un ciclo que contenga  $q_f$ , hay que hacer otro algoritmo para comprobar que sea accesible desde el estado final.