

Pseudocodice

I. Visite di grafi

I.1. Visita in ampiezza (Breadth-First Search)

Algoritmo *visitaInAmpiezza* (grafo $G = (V, E)$, vertice s) \rightarrow albero

```
/* Visita in ampiezza di un grafo non orientato connesso a partire da un
   vertice  $s$  e costruzione di un albero ricoprente ottenuto selezionando gli
   archi secondo l'ordine della visita */
 $C \leftarrow$  coda vuota
 $T \leftarrow (\{s\}, \emptyset)$  // albero formato dal solo vertice  $s$ 
marca  $s$  come raggiunto
 $C.enqueue(s)$ 
while not  $C.isEmpty()$  do
   $u \leftarrow C.dequeue()$ 
  foreach  $(u, v) \in E$  do
    if  $v$  non è marcato come raggiunto then
       $T \leftarrow (vertici(T) \cup \{v\}, archi(T) \cup \{(u, v)\})$ 
      // aggiunge il vertice  $v$  e l'arco  $(u, v)$  a  $T$ 
      marca  $v$  come raggiunto
       $C.enqueue(v)$ 
return  $T$ 
```

I.2. Visita in profondità (Depth-First Search)

Algoritmo *visitaInProfondità* (grafo $G = (V, E)$, vertice s) \rightarrow albero

```
/* Visita in profondità di un grafo non orientato connesso a partire da un
   vertice  $s$  e costruzione di un albero ricoprente ottenuto selezionando gli
   archi secondo l'ordine della visita */
 $T \leftarrow (\{s\}, \emptyset)$  // albero formato dal solo vertice  $s$ 
visitaRicorsiva( $G, s, T$ )
return  $T$ 
```

Procedura *visitaRicorsiva*(grafo $G = (V, E)$, vertice u , albero T)

```
marca  $u$  come visitato
foreach  $(u, v) \in E$  do
  if  $v$  non è marcato come visitato then
     $T \leftarrow (vertici(T) \cup \{v\}, archi(T) \cup \{(u, v)\})$ 
    // aggiunge il vertice  $v$  e l'arco  $(u, v)$  a  $T$ 
    visitaRicorsiva( $G, v, T$ )
```
