

```
// visitare tutti i nodi a distanza  $k$  prima di visitare i nodi a distanza  $k+1$ 
```

```
bfs(GRAPH  $G$ , NODE  $r$ )
```

```
    QUEUE  $S \leftarrow$  Queue
```

```
     $S.enqueue(r)$  // inserisco la radice
```

```
    // inizializzazione
```

```
    bool[]  $visitato \leftarrow$  bool[ $1 \dots G.n$ ]
```

```
    per ciascun  $u \in G.V - \{r\}$  fai  $visitato[u] \leftarrow$  falso
```

```
     $visitato[r] \leftarrow$  vero // radice visitata
```

```
    // visita del grafo
```

```
    finché not  $S.isEmpty$  fai
```

```
        NODE  $u \leftarrow S.dequeue$ 
```

```
        { esamina il nodo  $u$  }
```

```
        per ciascun  $u \in G.adj(u)$  fai
```

```
            { esamina l'arco  $(u, v)$  }
```

```
            se not  $visitato[v]$  allora
```

```
                 $visitato[v] \leftarrow$  vero
```

```
                 $S.enqueue$ 
```