

```
function breadth-first-search(vertices, source)
```

```
    frontier ← {source}
```

```
    next ← {}
```

```
    parents ← [-1, -1, ... -1]
```

```
    while frontier ≠ {} do
```

```
        ① BFS-bottom-up-step(vertices, frontier, next, parents)
```

```
        frontier ← next
```

```
        next ← {}
```

```
    end while
```

```
    return tree
```

---

```
function BFS-bottom-up-step(vertices, frontier, next, parents)
```

```
    ② for v ∈ graphCSR do
```

```
        ③ if parents[v] = -1 then
```

```
            ④ for n ∈ inNeighbor[v] do
```

```
                ⑤ if n ∈ frontier do
```

```
                    parents[v] ← n
```

```
                    ⑥ next ← next ∪ {v}
```

```
                    ⑦ break
```

```
                end if
```

```
            end for
```

```
        end if
```

```
    end for
```

②

Sequential access, each cacheline has high reuse

⑦ ⑤ ③

load based, each frontier different bitmap. Custom Branches or Filter out scheduled vertices

⑥ ④

Random Access Fetch Vertex Property Data