

G.DFS(A)

```
for each vertex v
    v.visited = false
    v.start   = infty
    v.finish  = infty
    v.parent  = null
```

time = 0

DFS-visit(A)

```
for each vertex x
    if (x is not visited) DFS-visit(x)
```

DFS-visit(vertex v) {

```
    v.visited = true
    time++
    v.start = time
```

```
    for each neighbor w of v {
```

```
        if (w is not visited) {
            w.parent = v
            DFS-visit(w)
```

```
        else {
```

```
            if (w.start < v.start && w.finish == infinity) // back edge
                // whatever needs to be done to handle back edges
                // e.g., for cycle checking, return false
```

```
            if (w.start > v.start) // forward edge
                // whatever needs to be done to handle back edges
```

```
            if (w.start < v.start && w.finish != infinity) // cross edge
                // whatever needs to be done to handle back edges
```

```
        }
```

```
    time++
    v.finish = time
```

```
}
```