Same Check and Marking for Lower Child (Value 8)

```
    if (0 == (maze[x][y] & 4)) {
        can_reach (x, y - 1);
    }
    if (0 == (maze[x][y] & 8)) {
        can_reach (x, y + 1);
    }
    if (0 != (maze[x][y] & 16)) {
        saw_exit = 1;
    }
}
```

No lower wall?

Space below is reachable.

Finally, Check and Mark Exit (Value 16)

```
    if (0 == (maze[x][y] & 4)) {
        can_reach (x, y - 1);
    }
    if (0 == (maze[x][y] & 8)) {
        can_reach (x, y + 1);
    }
    if (0 != (maze[x][y] & 16)) {
        saw_exit = 1;
    }
}
```

Exit here?

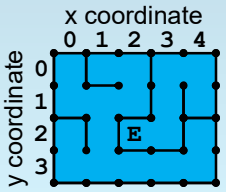
Record having seen exit.

Does Our `can_reach` Function Answer My Question?

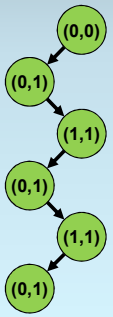
How do we use **`can_reach`** to answer my question about getting from (0,0) to the exit?

1. Fill **`found`** and **`saw_exit`** with 0s.
2. Call **`can_reach (0,0)`**.
3. Check **`saw_exit`**.

Does it work?



Maybe There's a Bug?



Let's try it!

(Are you still mad about my asking you to write all of Fibonacci?)