Outline

1. Example
2. DFS vs. BFS
3. Another Example
Outline

1. Example
2. DFS vs. BFS
3. Another Example
Example exam question

Prove or disprove:

d) \(2^{n+1} = O(2^n)\)

e) \(\log(f(n)) = O(\log(g(n))) \implies f(n) = O(g(n))\)

f) \(\max(f(n), g(n)) = \Theta(f(n) + g(n))\)
Outline

1. Example

2. DFS vs. BFS

3. Another Example
Graphs - what are they good for?
Provide a good representation of...

- Data structures (linked lists, hash tables, trees)
- The internet
- DNA
- A road network
- A sewer system
- The circulation of your favorite bodily fluid
- ...

Google uses graphs! I use graphs!
And then?
With the representation, we can isolate interesting properties of the graph, and thus discover interesting properties of the underlying system.

Example: A road system is abstracted as a directed graph in the natural way. What information would computing the strongly connected components of this graph give us?
And then?

With the representation, we can isolate interesting properties of the graph, and thus discover interesting properties of the underlying system.

**Example**: A road system is abstracted as a directed graph in the natural way. What information would computing the strongly connected components of this graph give us?

- Whether there is a path from every junction to all other junctions.
Fundamentals: DFS and BFS

- We look a vertex at a time, examining it and discovering its neighbours (moving the frontier).
- The difference lies in the order we explore the neighbours!
DFS

- When examining a node, just tag it as discovered and examine all not yet discovered neighbours first.
- So we examine the nodes using a LIFO policy.
BFS

- When examining a node, put all not yet discovered neighbours on hold and let them “wait for their turn”.
- So we examine the nodes using a FIFO policy.
The essence

- DFS uses LIFO, BFS uses FIFO.
- This means they are essentially the same, except that DFS uses a stack and BFS uses a queue.

Note that this glosses over a lot of important points, read the book for more detailed explanation.
Outline

1. Example
2. DFS vs. BFS
3. Another Example
Example exam question

Binary-Search($A, start, stop, key$)

1. ▶ Calculate the middle of the search-block
2. $i \leftarrow start + \lceil (stop - start)/2 \rceil$
3. ▶ If we have failed in our search, we return a negative index
4. if $start = stop$ and $A[i] \neq key$
5. then return $-1$
6. ▶ If we have found the key, we return its index
7. if $A[i] = key$
8. then return $i$
9. ▶ Otherwise, we recurse in the correct part of the array
10. if $A[i] > key$
11. then Binary-Search($A, start, i - 1, key$)
12. else Binary-Search($A, i + 1, stop, key$)

Derive the recurrence and its upper bound.