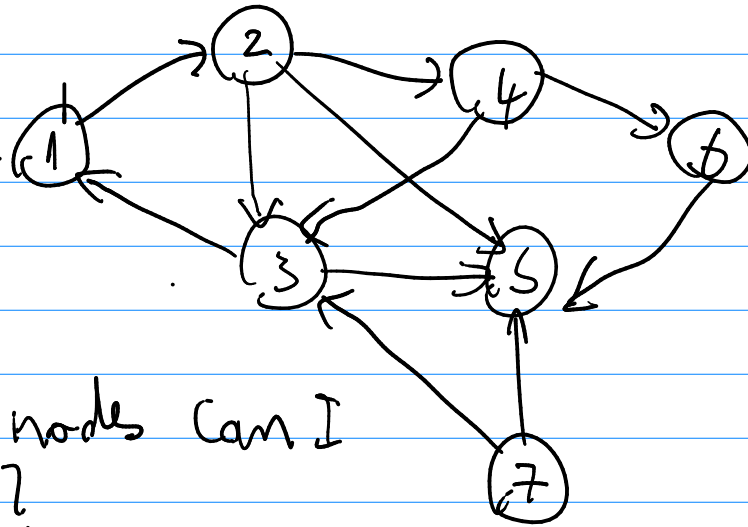


# Search

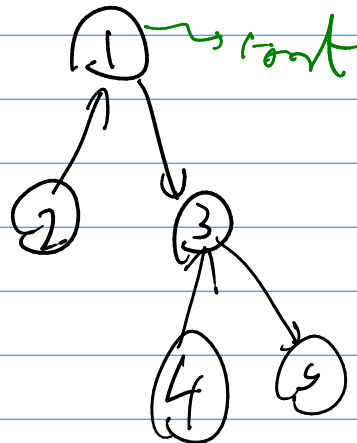
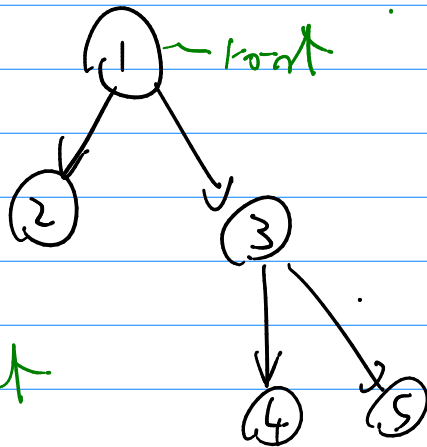


Which nodes can I reach?

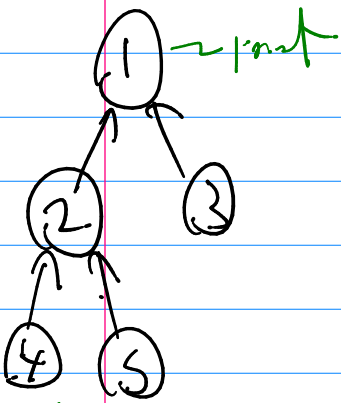
How?

But first, what is tree?

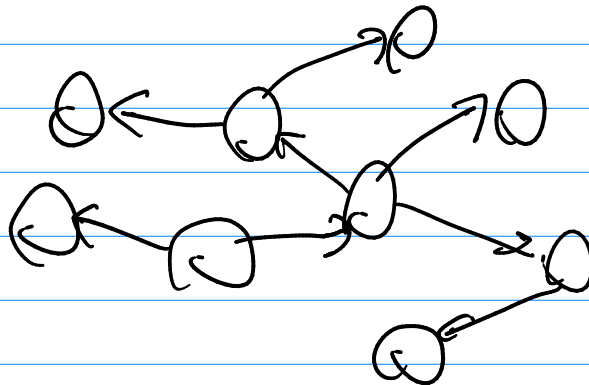
Directed  
out



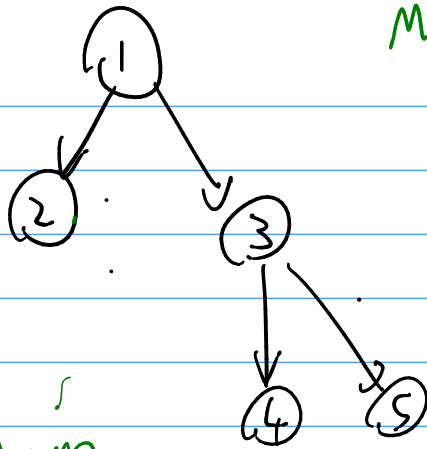
Rooted tree  
but not  
directed-in  
or out.



Directed-in



Tree  
not direct in  
or out  
with no root



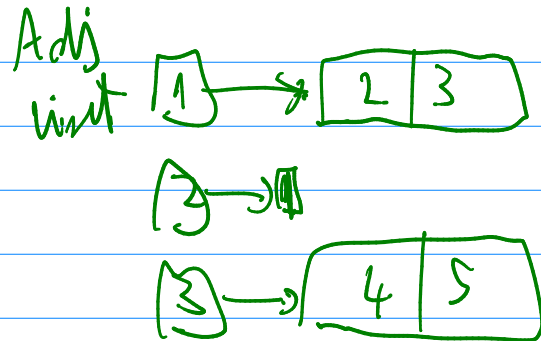
Adj  
Matrix

|   | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 1 | 1 |
| 4 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 |

Pred Array

Pred =

|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 1 | 1 | 3 | 3 |
| 1 | 2 | 3 | 4 | 5 |



Back to Search

algorithm search;  
begin

unmark all nodes in  $N$ ;

mark node  $s$ ;

pred( $s$ ) = 0;

next = 1;

order( $s$ ) =  $s$ ;

LIST = { $s$ }

while LIST  $\neq \emptyset$  do

begin

select a node  $i$  in LIST;

If node  $i$  is incident to an admissible arc  $(i, j)$  then

begin

mark node  $j$ ;

pred( $j$ ) =  $i$ ;

next = next + 1;

order( $j$ ) = next;

add node  $j$  to LIST;

end

else delete node  $i$  from LIST;

end;

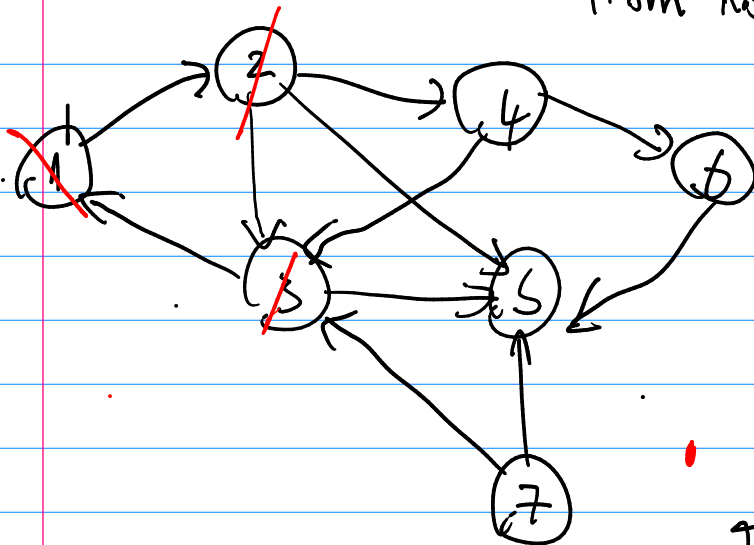
end;

marked = [0, 0, 0, 0, 0]

→ forget about those for now

Figure 3.4 Search

Find the breadth first tree starting from node 1.



$S = 1$

Initialization

marked = [1]

LIST = [1]

pred = [0, 0, 0, 0, 0]

• Select 1

I have an admissible arc (1, 2).

pred = [0, 1, 0, 0, 0]

marked = [1, 2]

LIST = [1, 2]

• Select 1

LIST = [2]

• Select 2

admissible arc (2, 3)

pred = [0, 1, 2, 0, ...]

marked = [1, 2, 3]

LIST = [2, 3]

• Select 2

admissible arc (2, 4)

pred = [0, 1, 2, 2, ...]

marked = [1, 2, 3, 4]

LIST = [2, 3, 4]

# Breath first