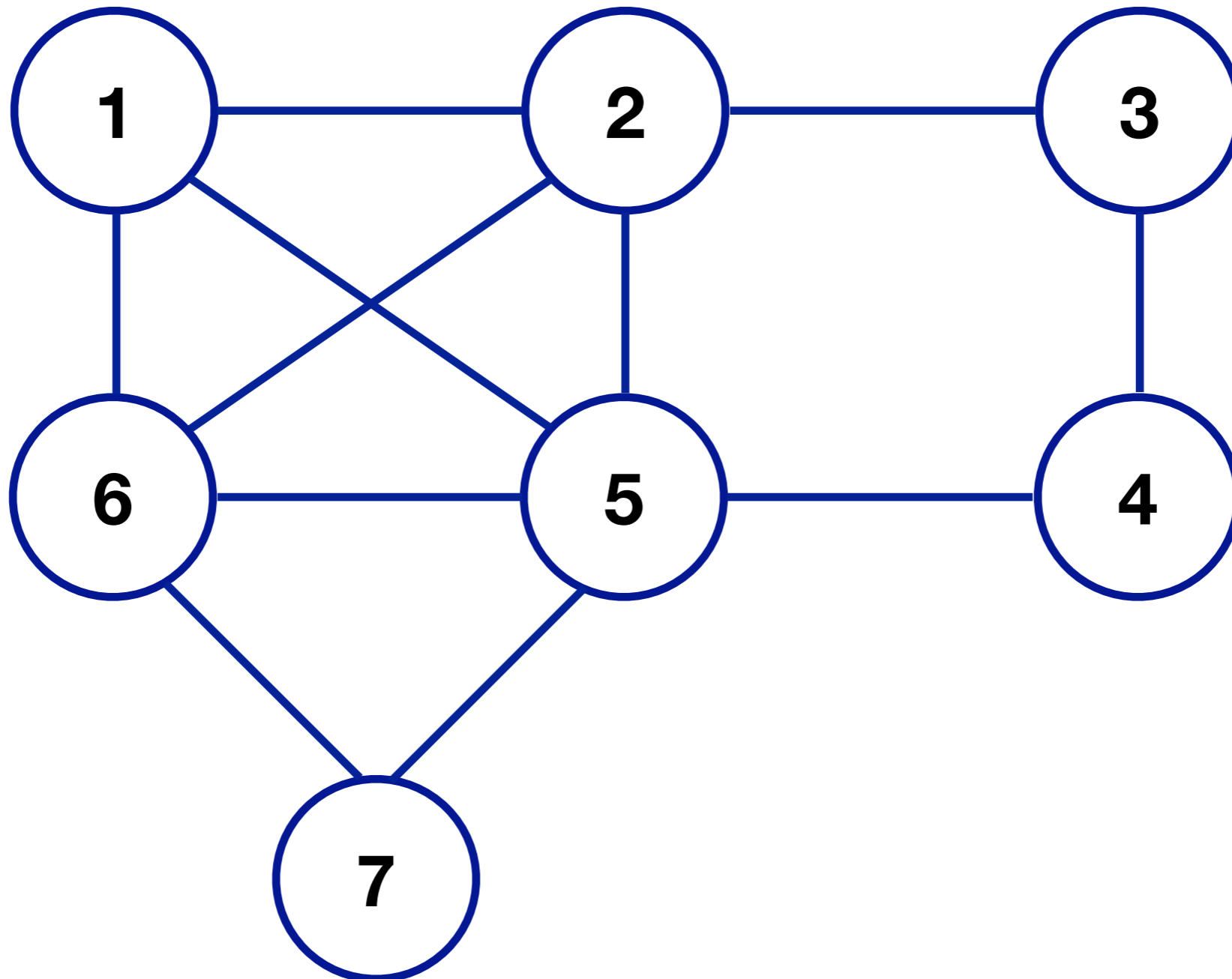

Depth-First Search / Traversal



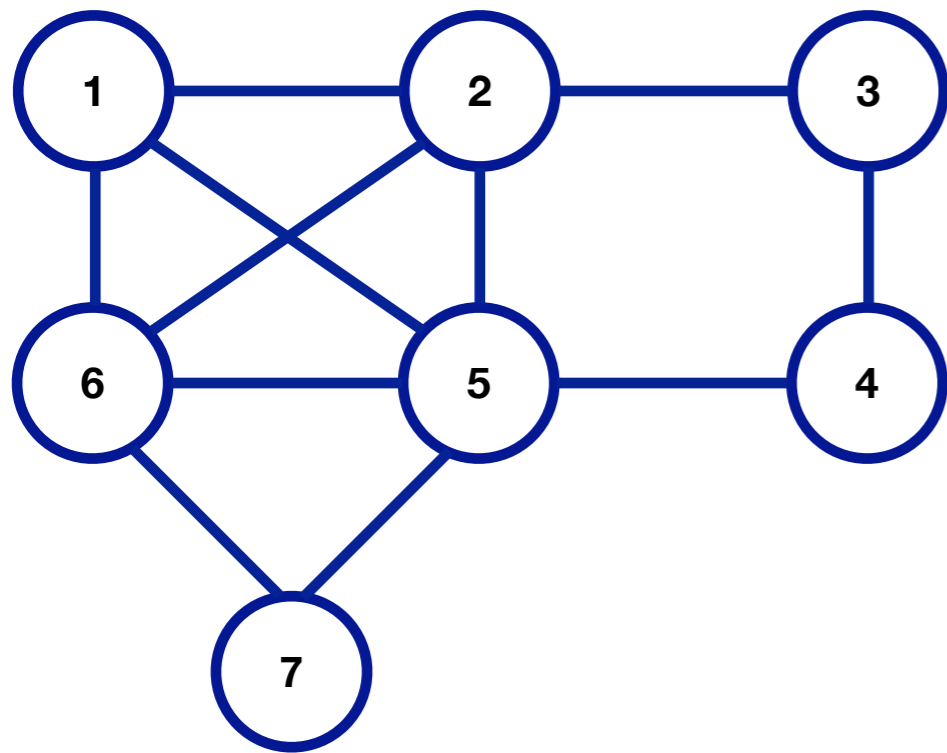
Alan G. Labouseur, Ph.D.
Alan.Labouseur@Marist.edu

Graphs

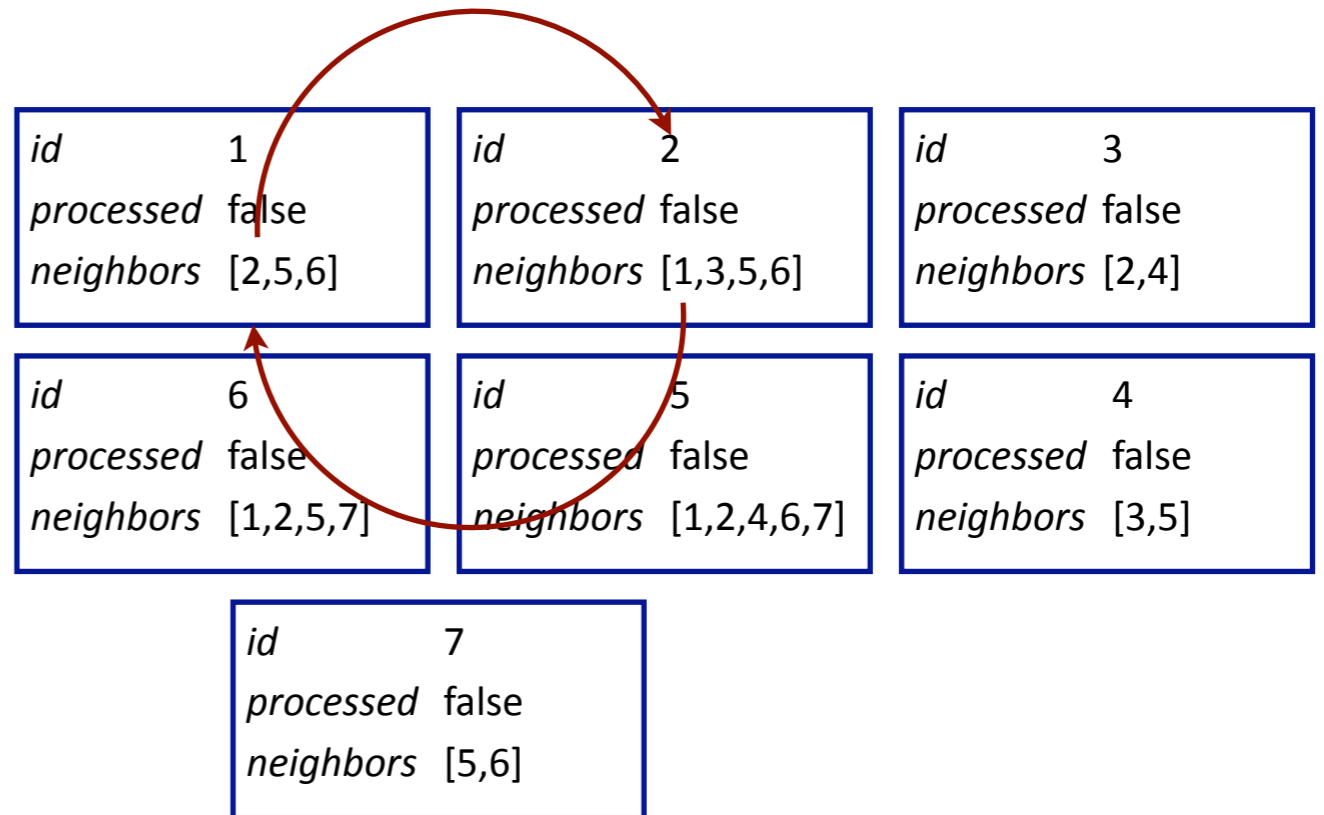


Graphs

Graph . . .

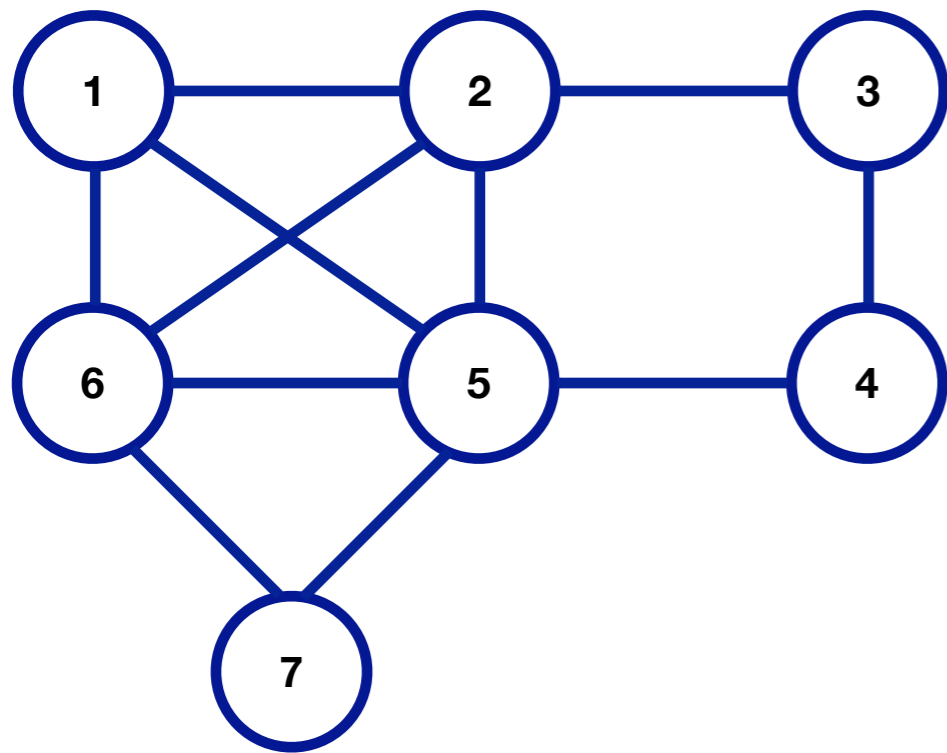


as Linked Objects

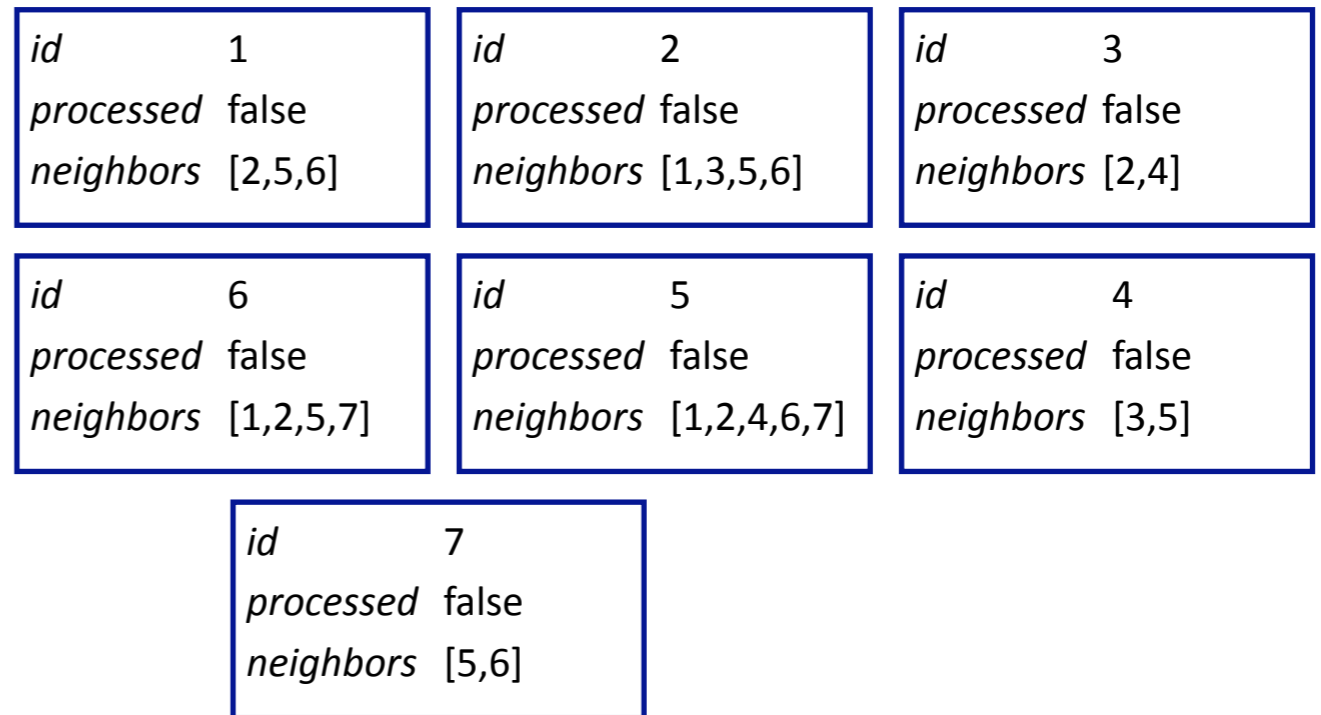


Graphs

Graph . . .



as Linked Objects

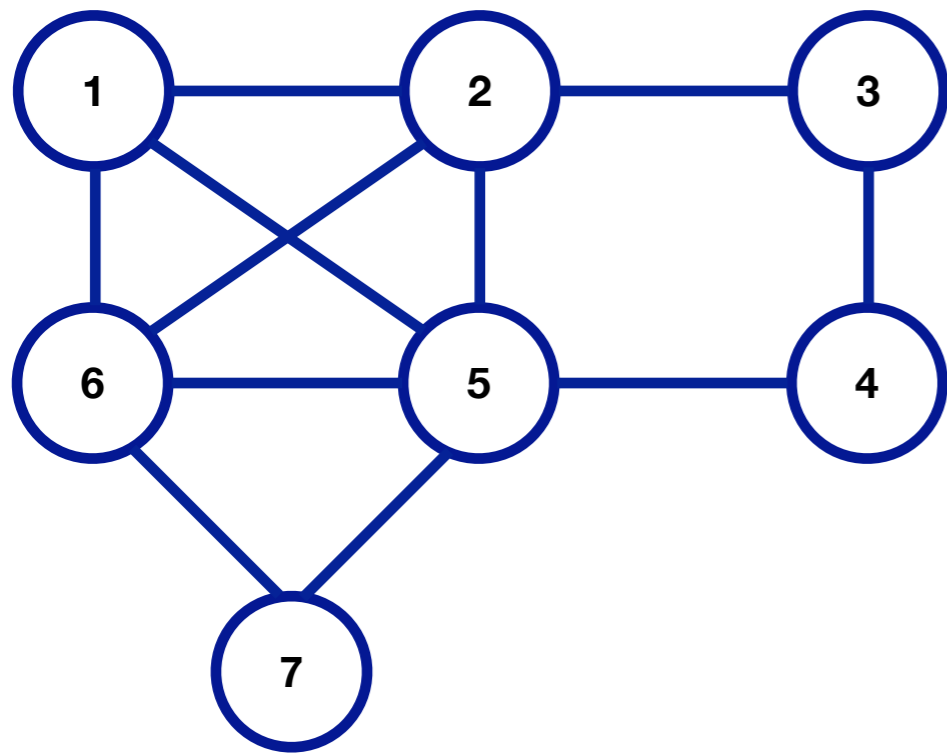


Adjacency List

```
[1] 2 5 6  
[2] 1 3 5 6  
[3] 2 4  
[4] 3 5  
[5] 1 2 4 6 7  
[6] 1 2 5 7  
[7] 5 6
```

Graphs

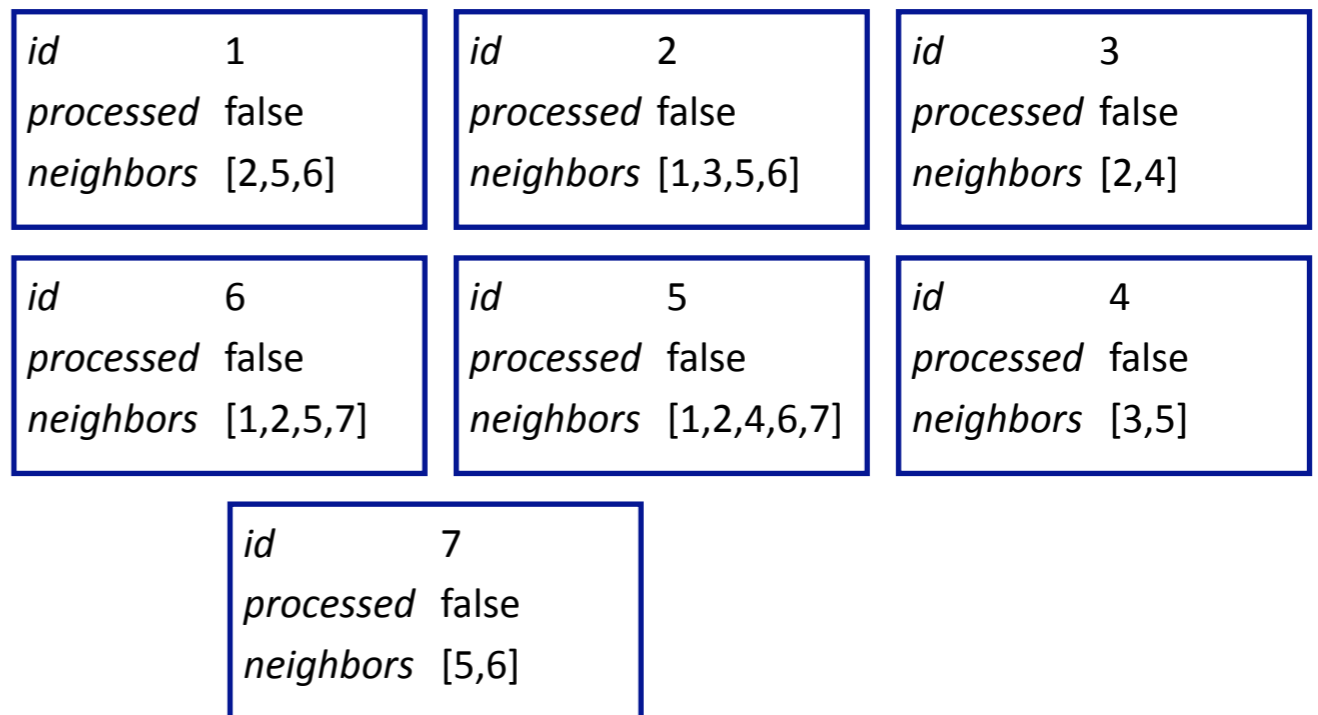
Graph . . .



Adjacency
List

```
[1] 2 5 6  
[2] 1 3 5 6  
[3] 2 4  
[4] 3 5  
[5] 1 2 4 6 7  
[6] 1 2 5 7  
[7] 5 6
```

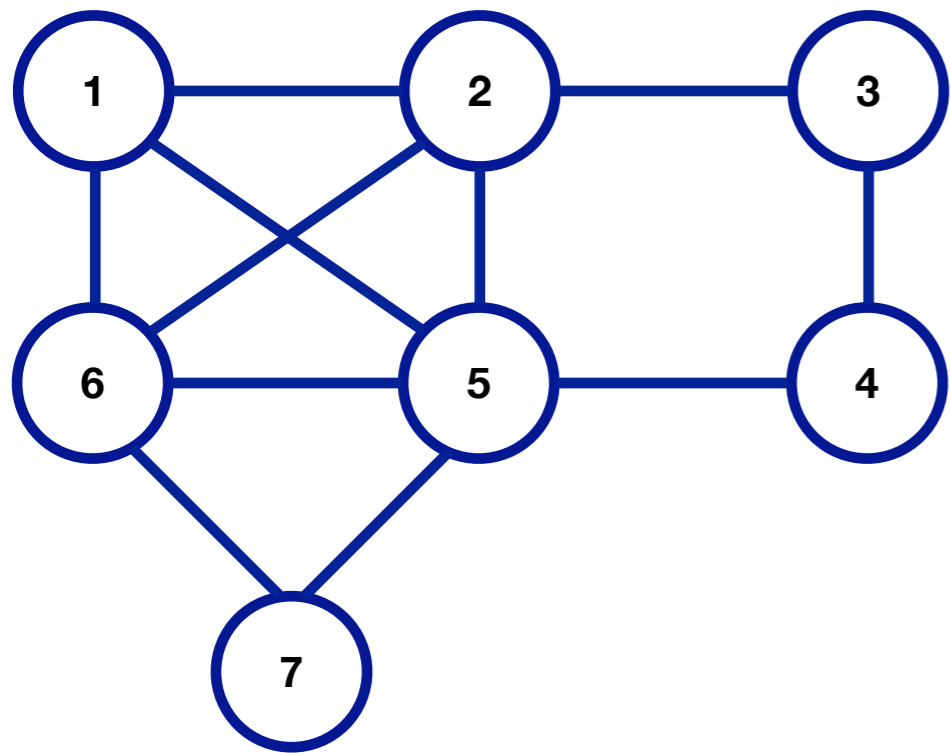
as Linked Objects



Matrix

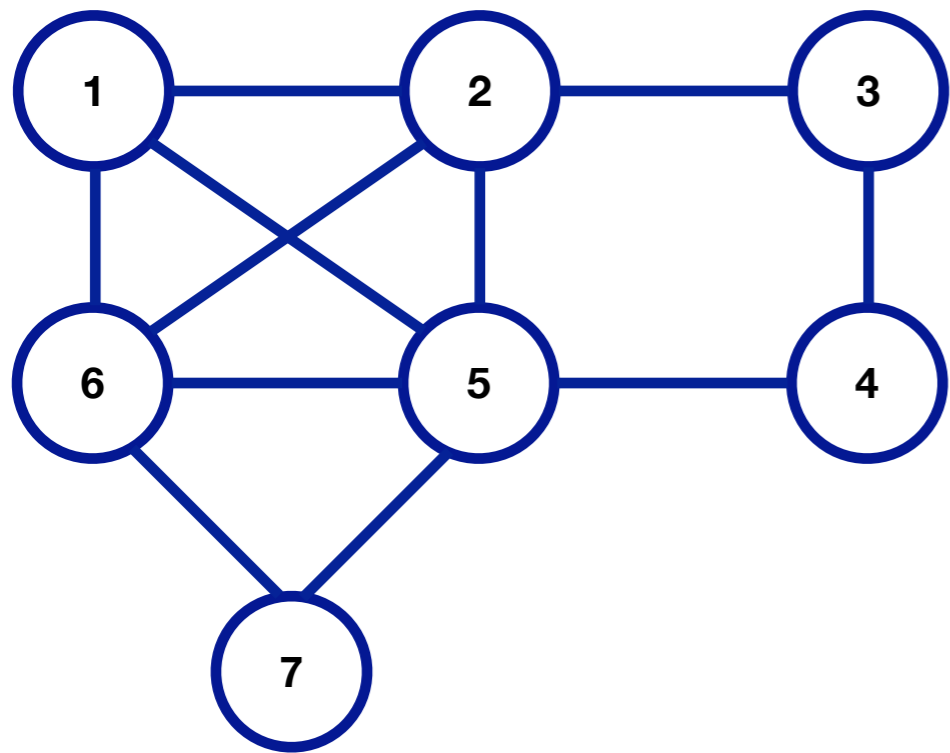
```
  1 2 3 4 5 6 7  
1 . 1 . . 1 1 .  
2 1 . 1 . 1 1 .  
3 . 1 . 1 . . .  
4 . . 1 . 1 . .  
5 1 1 . 1 . 1 1  
6 1 1 . . 1 . 1  
7 . . . . 1 1 .
```

Depth-First Search / Traversal



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

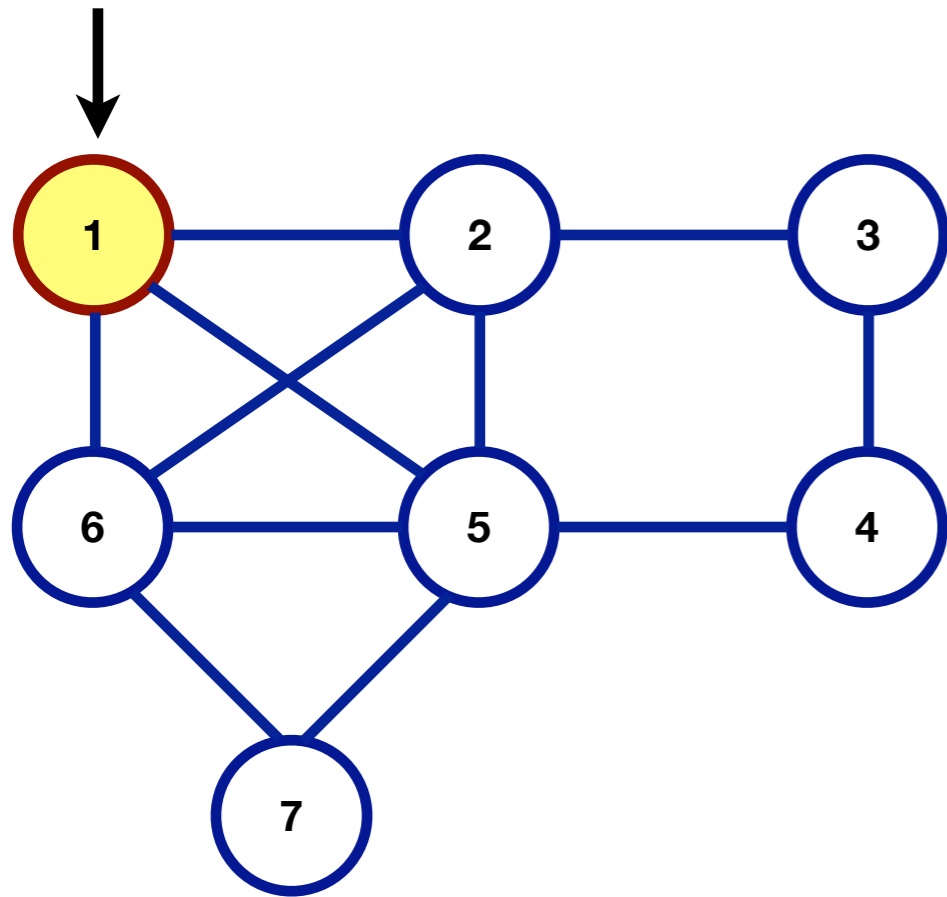


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Recursion.
Yay!

Depth-First Search / Traversal

myGraph.DFS(vertex1)

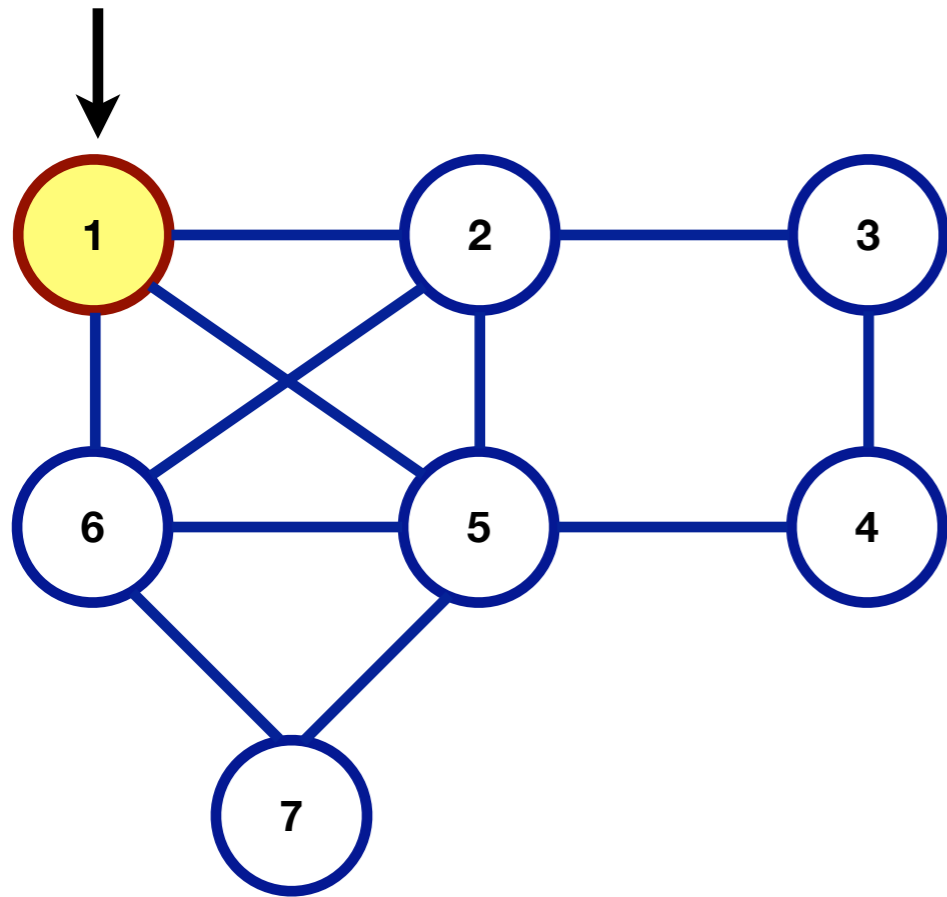


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)

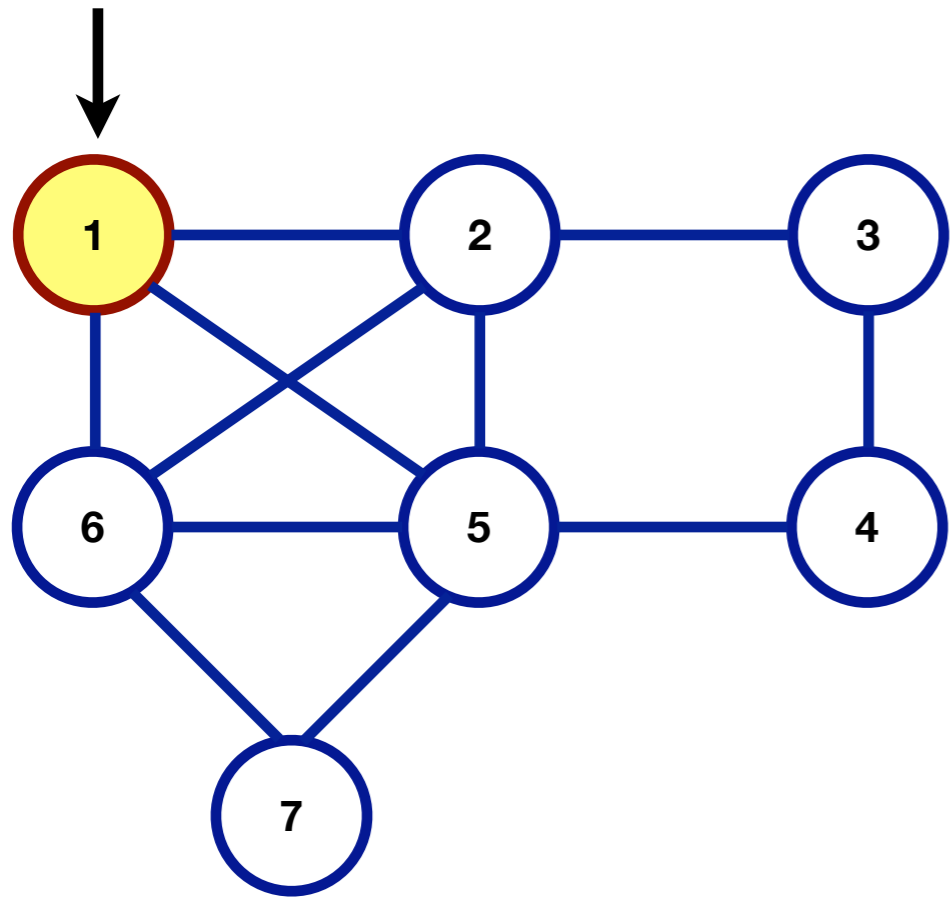


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



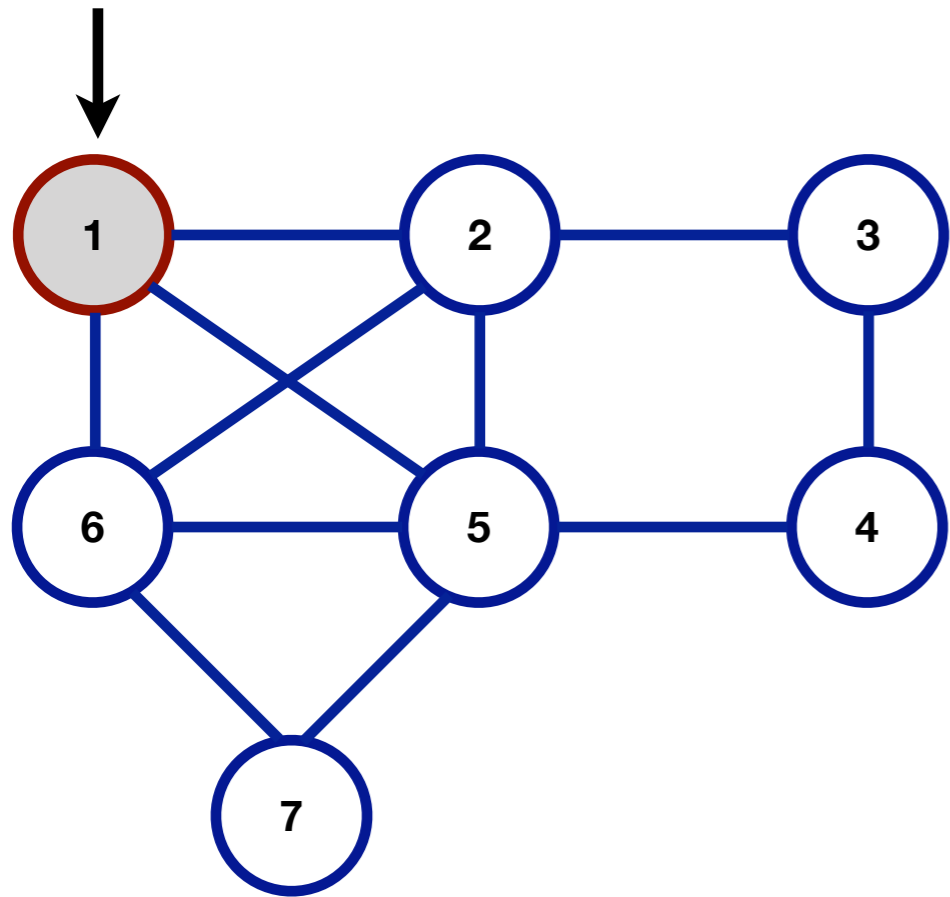
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



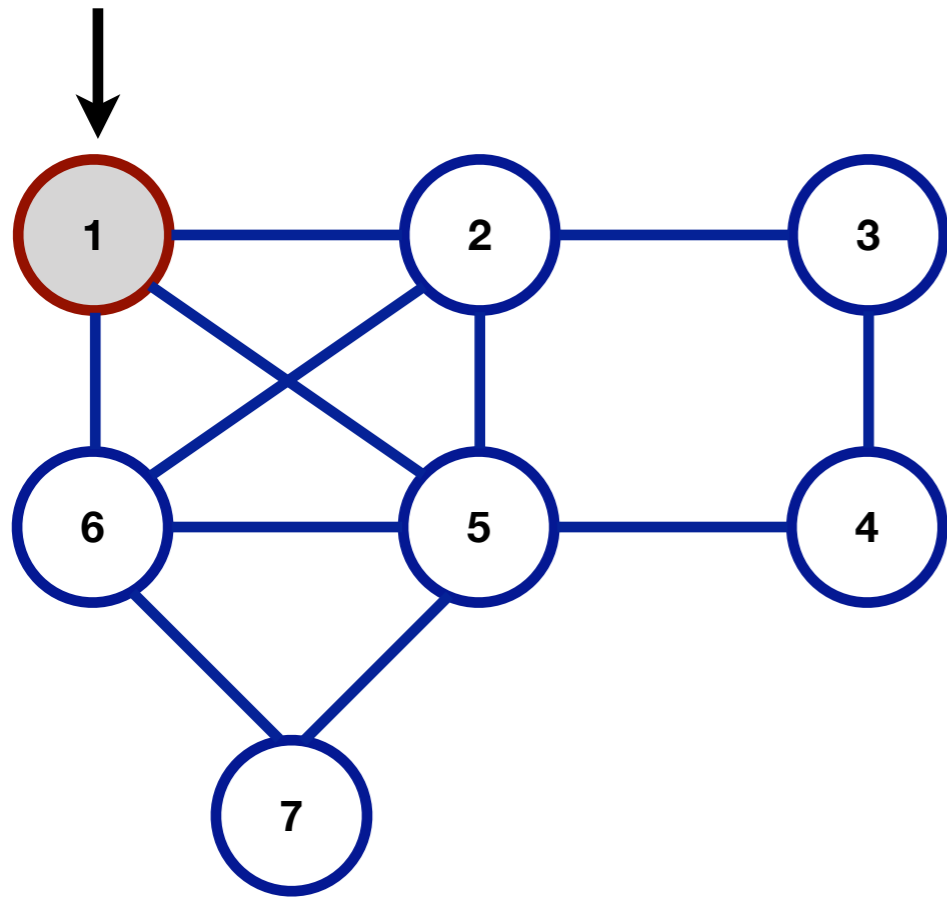
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



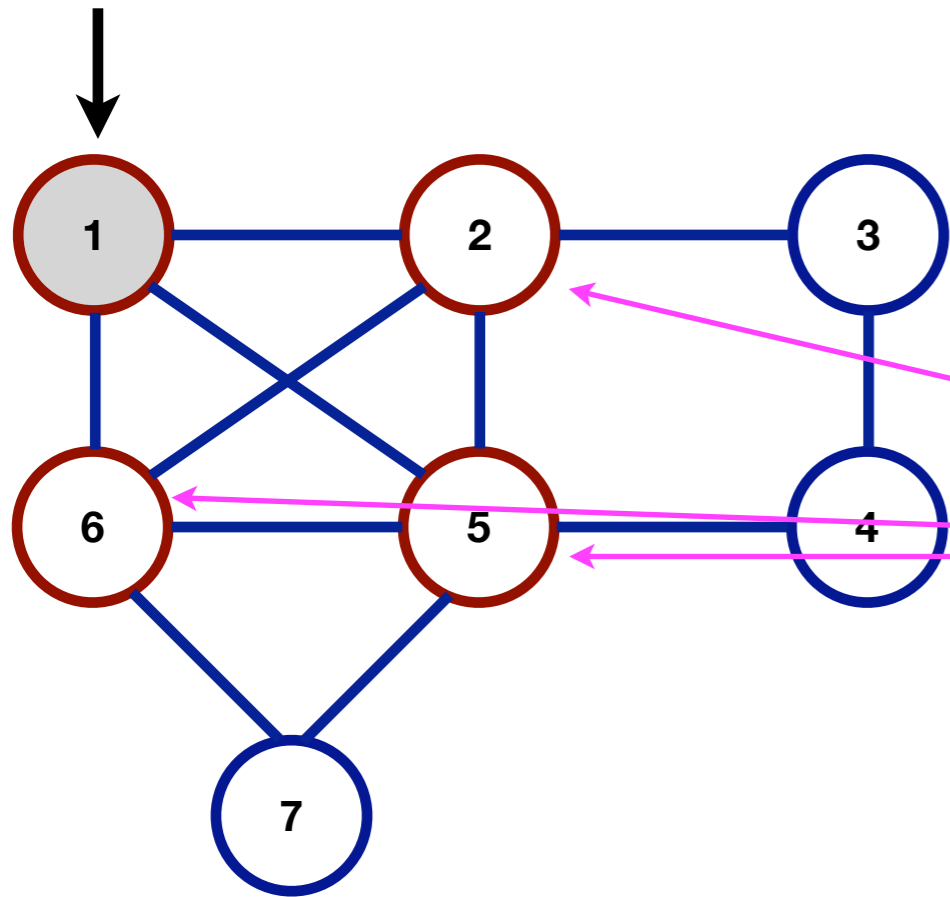
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

```
[1] 2 5 6
[2] 1 3 5 6
[3] 2 4
[4] 3 5
[5] 1 2 4 6 7
[6] 1 2 5 7
[7] 5 6
```

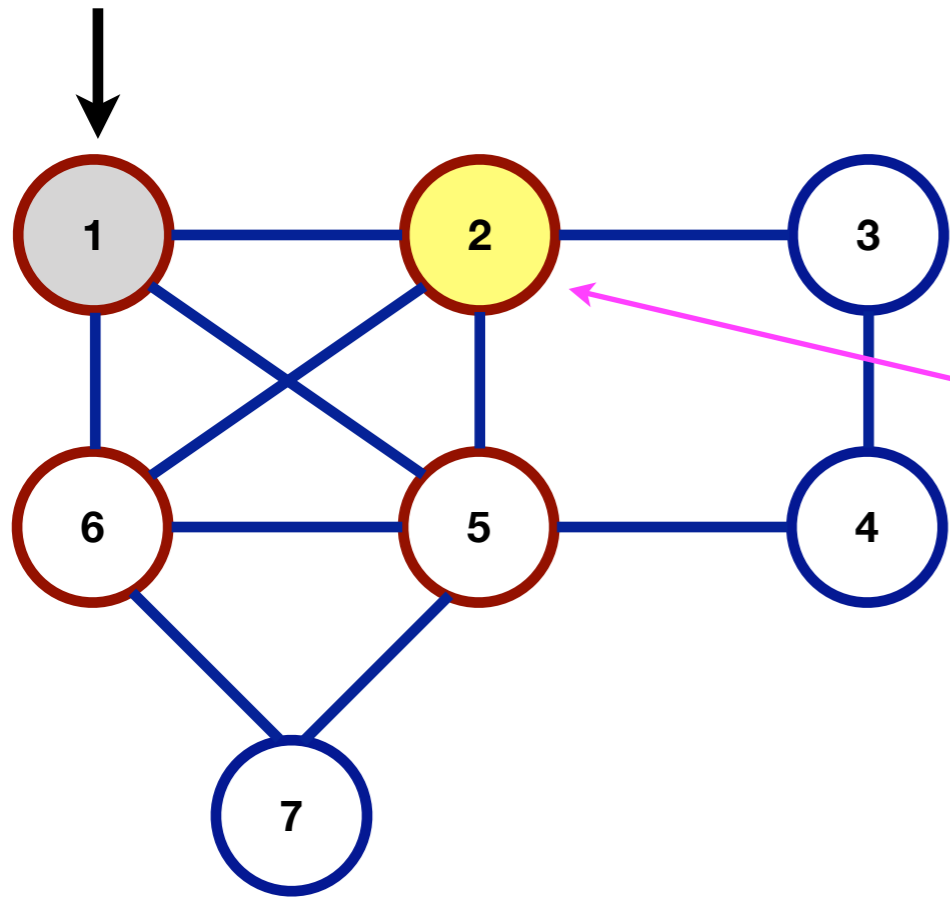
Adjacency List

1

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



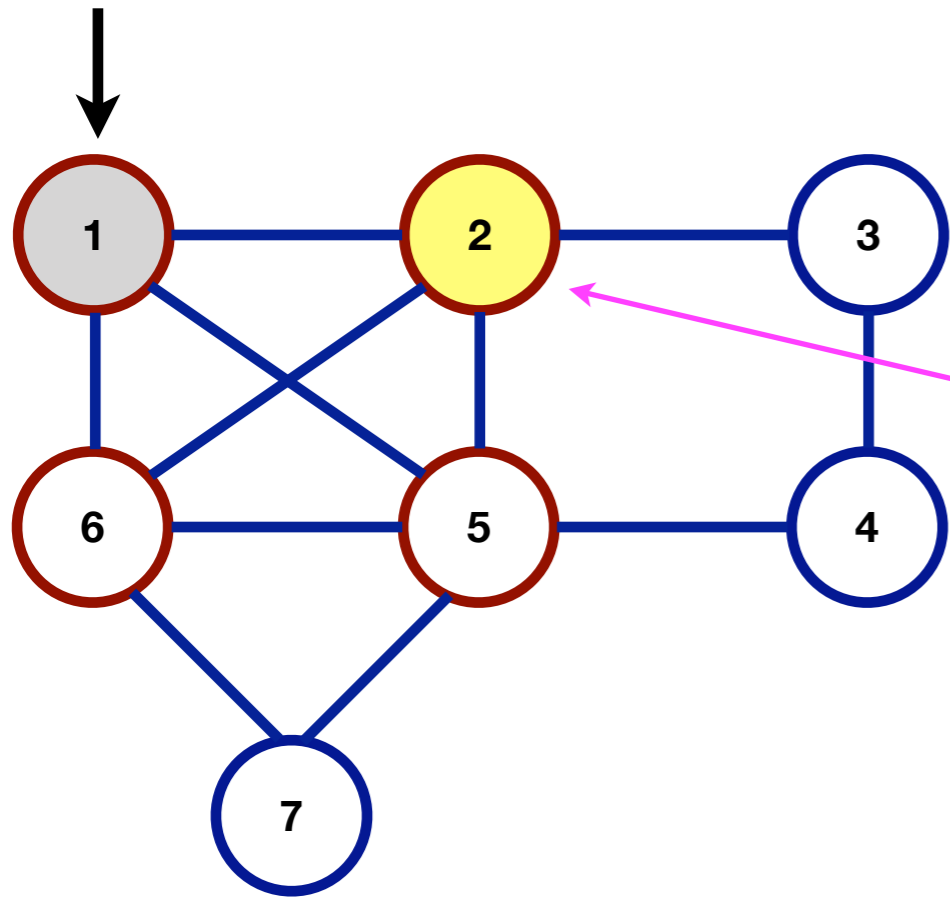
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1

Runtime Stack

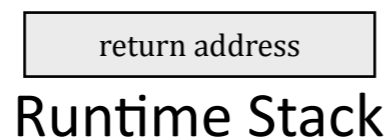
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

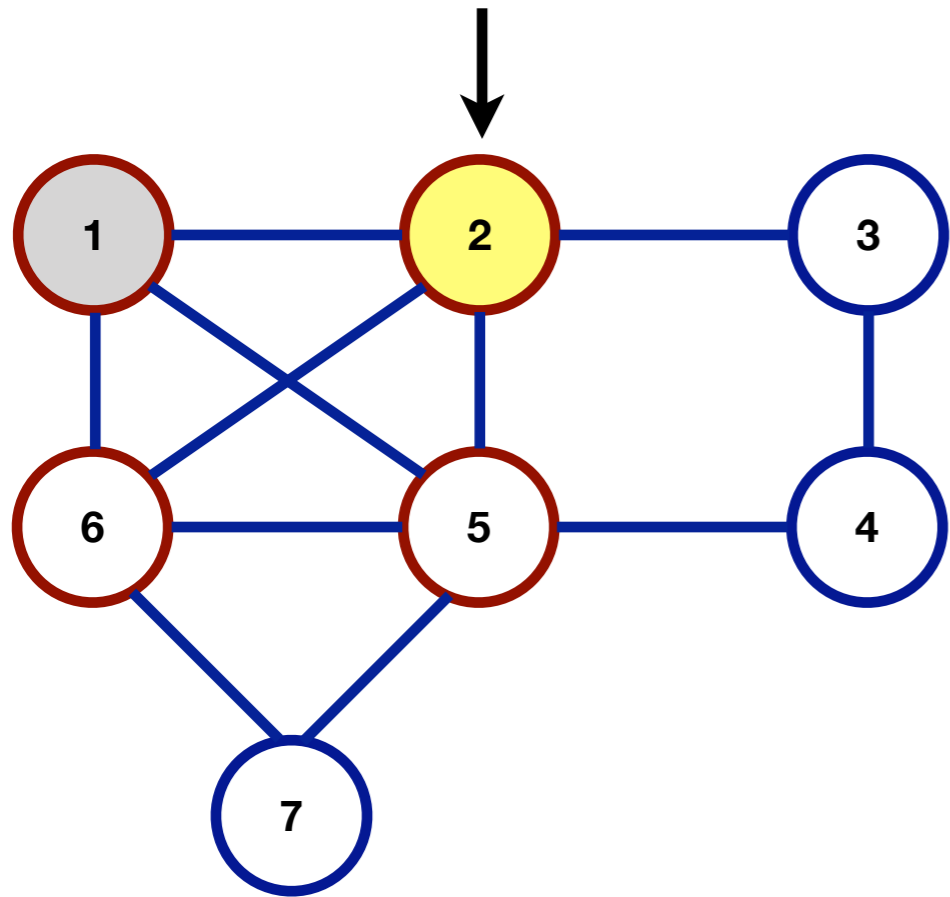
1



Recursive call.
Push return address
on the runtime stack.

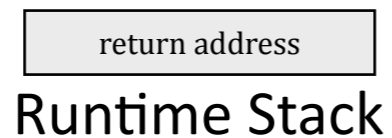
Depth-First Search / Traversal

myGraph.DFS(vertex1)



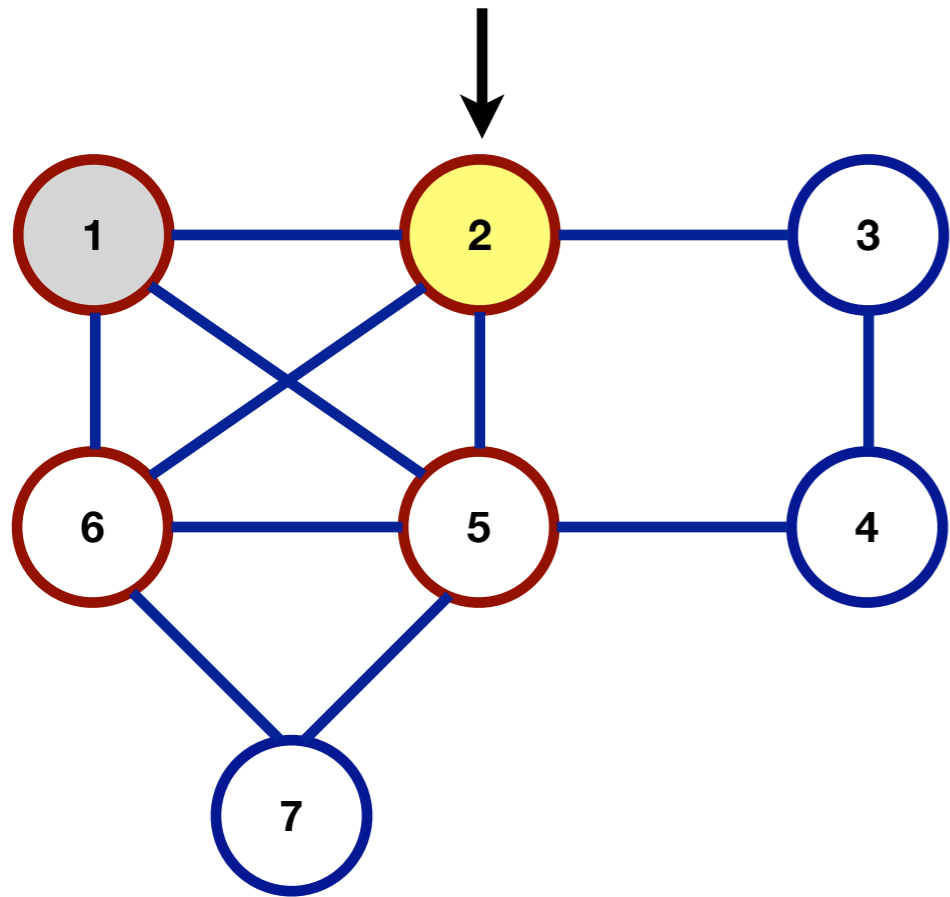
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1



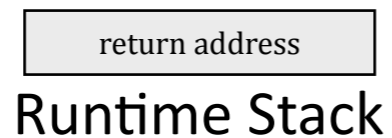
Depth-First Search / Traversal

myGraph.DFS(vertex1)



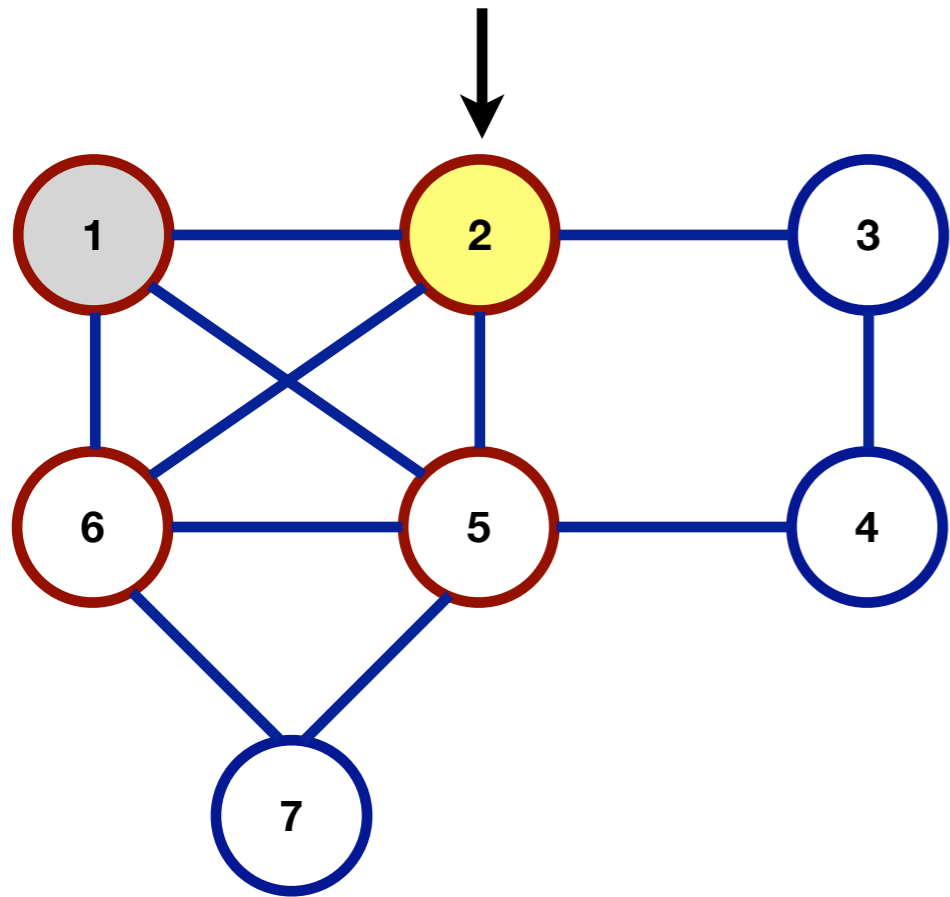
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1



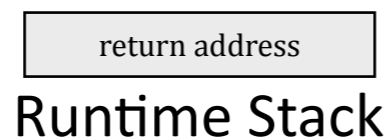
Depth-First Search / Traversal

myGraph.DFS(vertex1)



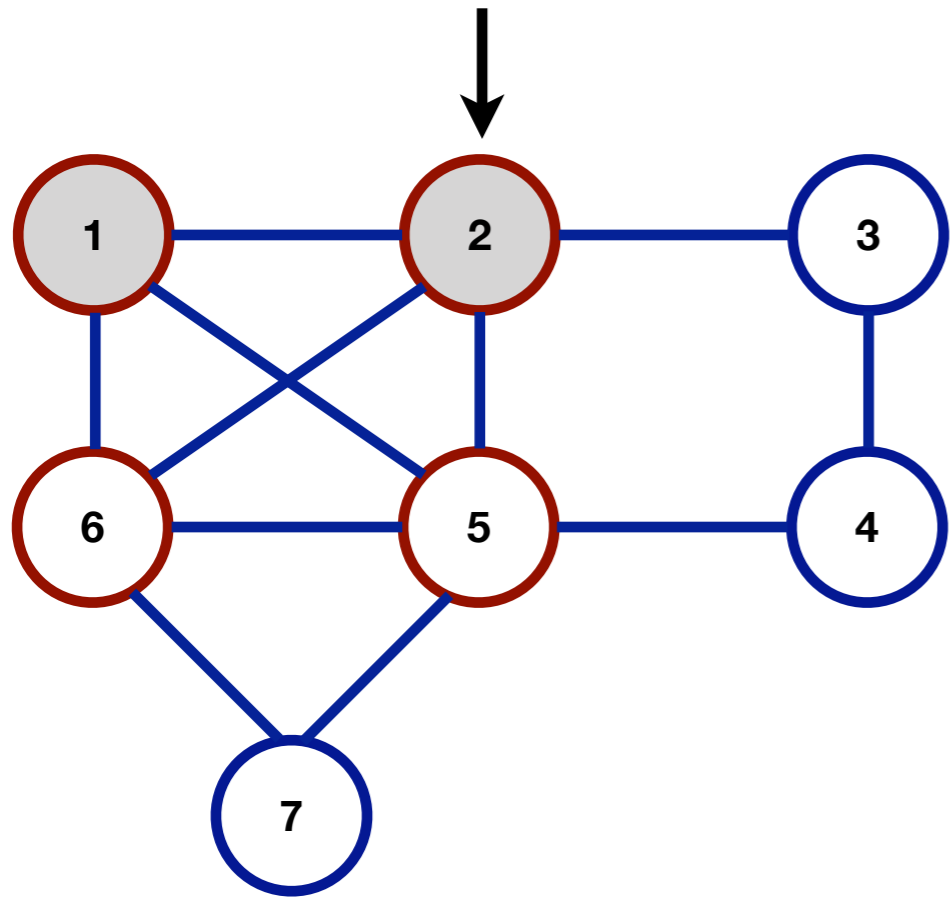
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2



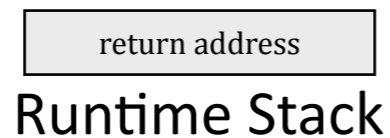
Depth-First Search / Traversal

myGraph.DFS(vertex1)



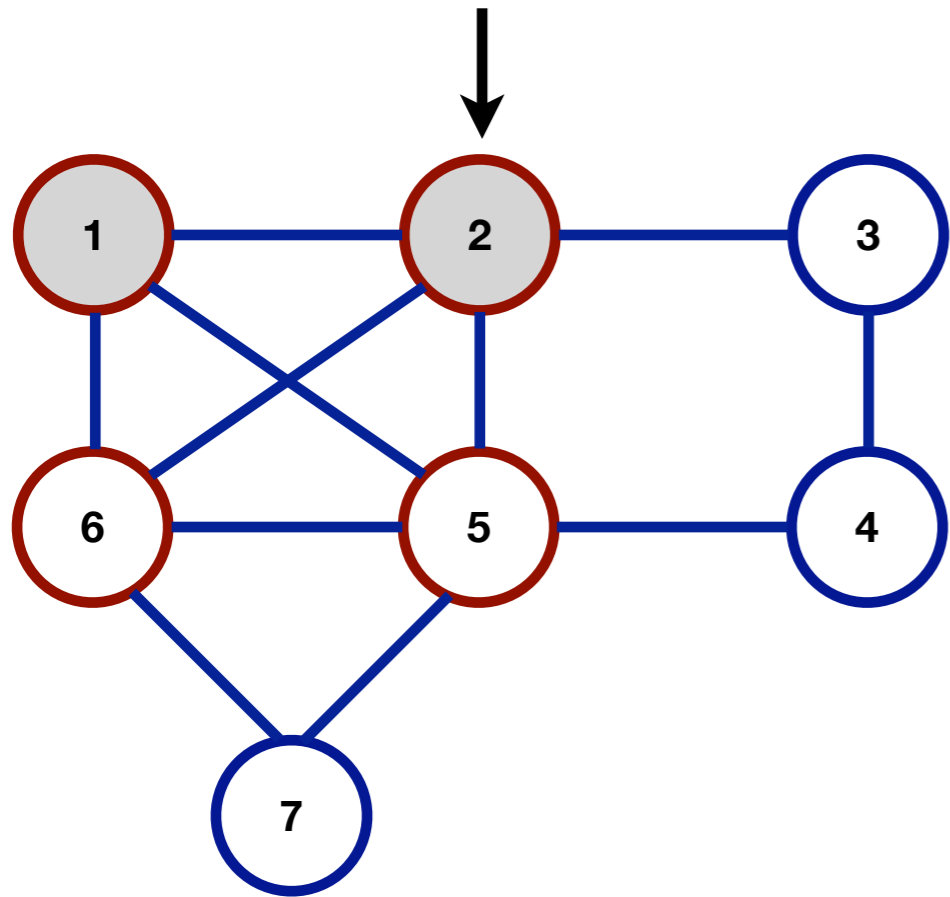
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2



Depth-First Search / Traversal

myGraph.DFS(vertex1)



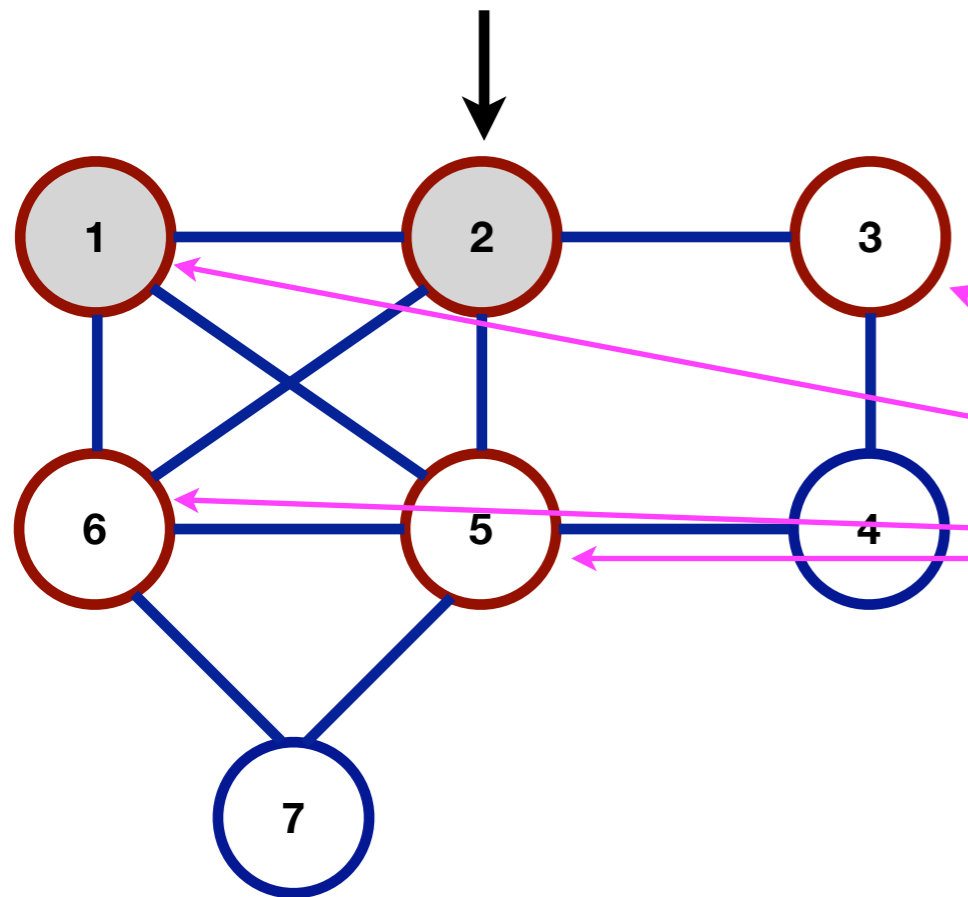
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)

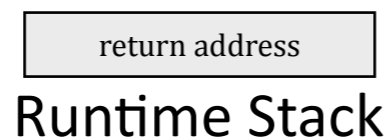


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

[1]	2	5	6		
[2]	1	3	5	6	
[3]	2	4			
[4]	3	5			
[5]	1	2	4	6	7
[6]	1	2	5	7	
[7]	5	6			

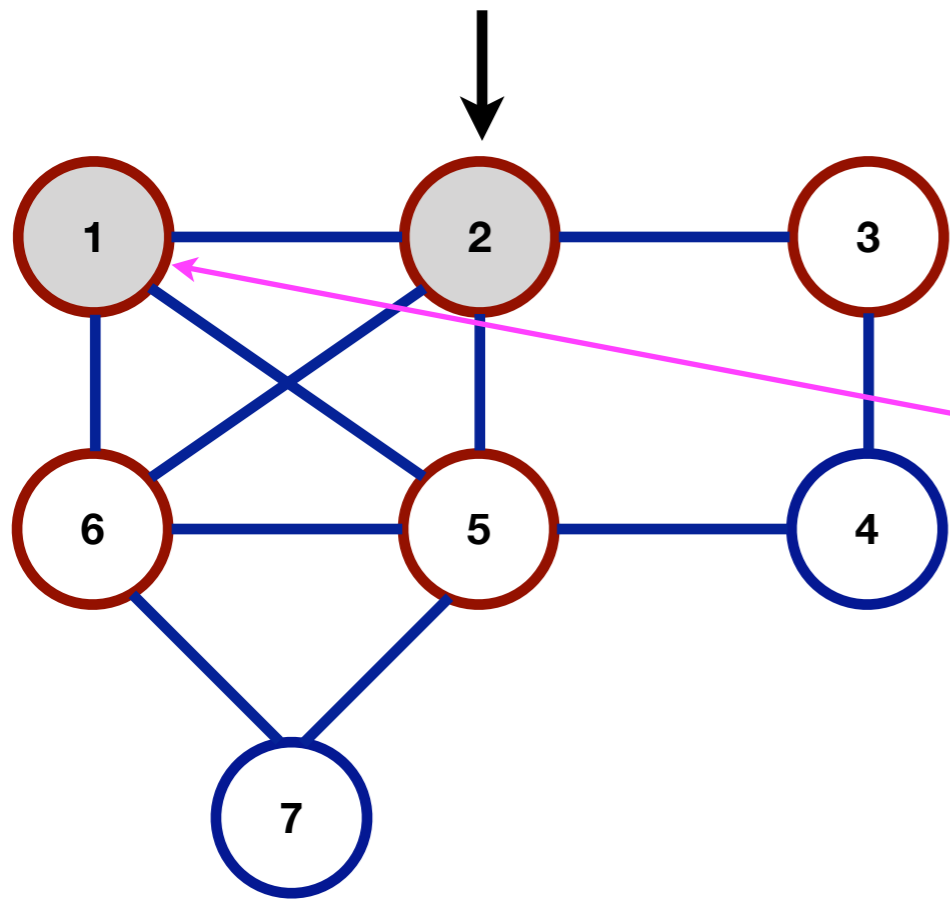
Adjacency List

1 2



Depth-First Search / Traversal

myGraph.DFS(vertex1)



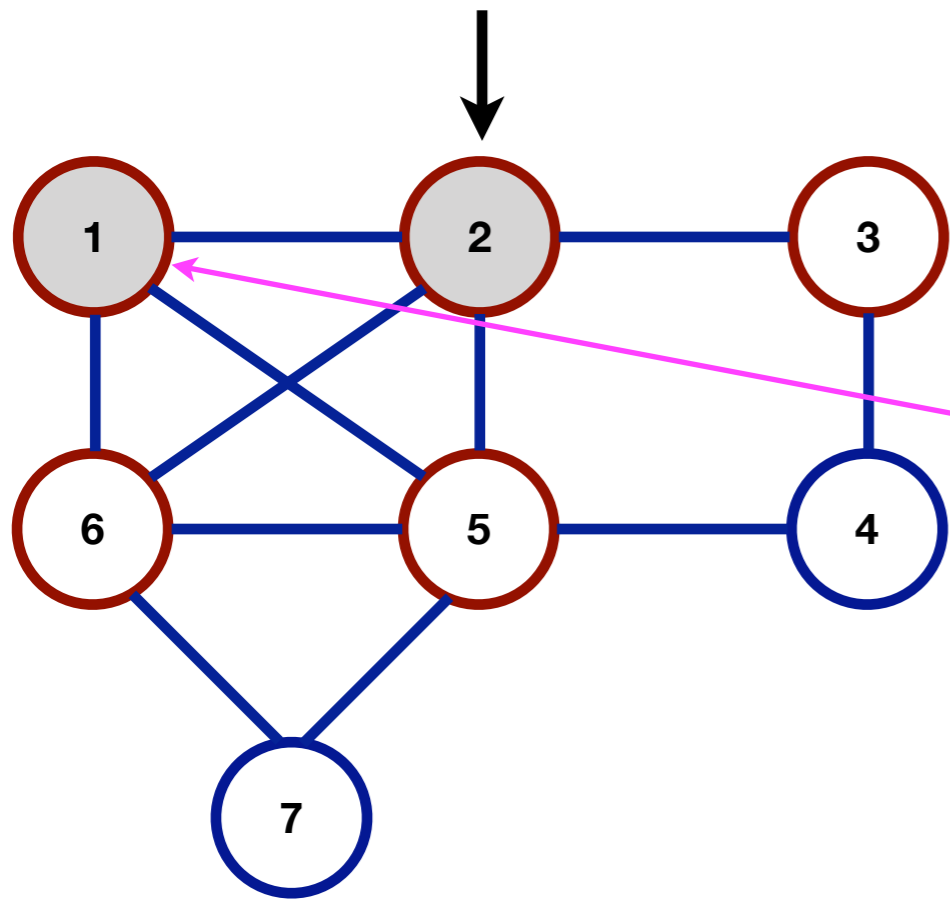
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2

return address
Runtime Stack

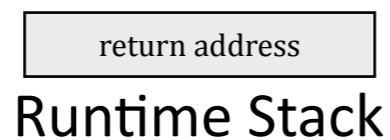
Depth-First Search / Traversal

myGraph.DFS(vertex1)



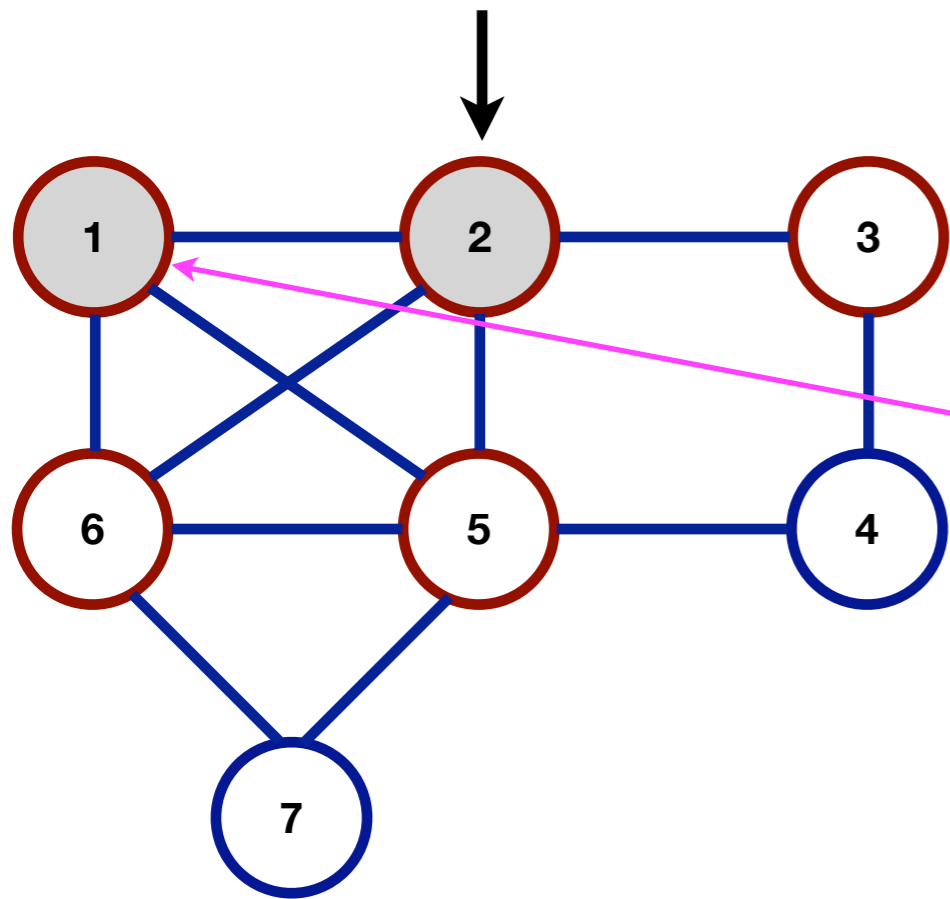
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2



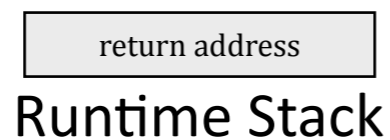
Depth-First Search / Traversal

myGraph.DFS(vertex1)



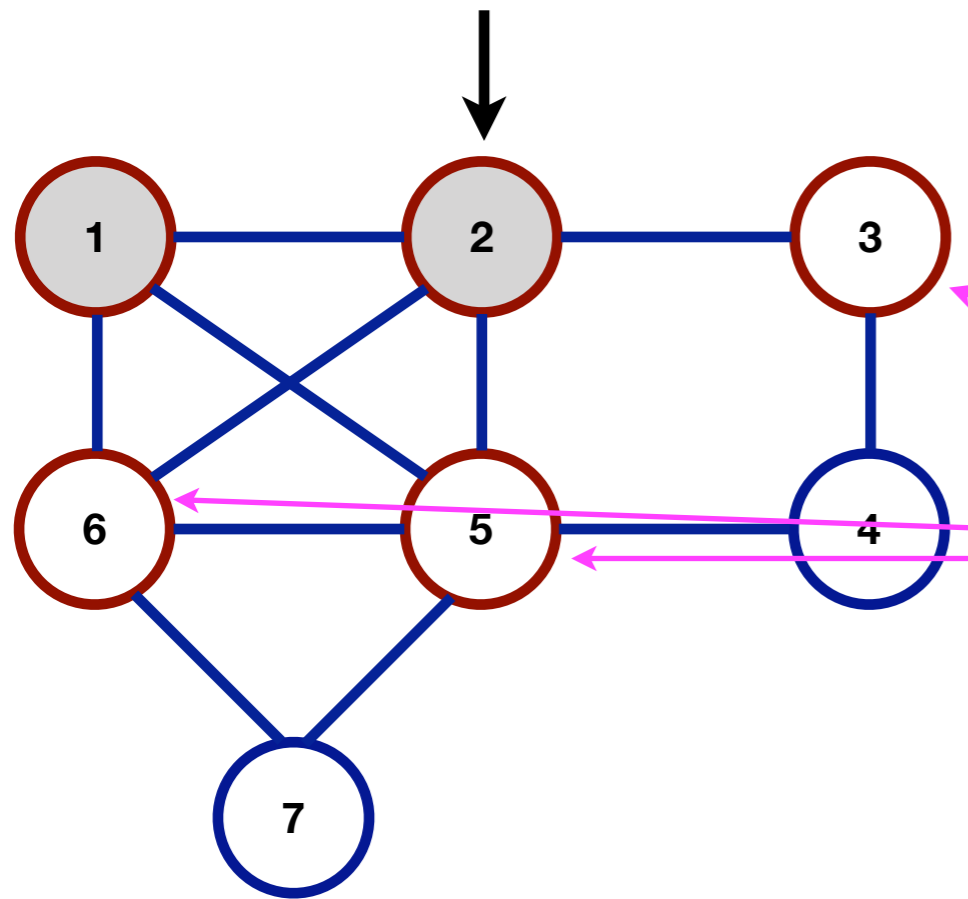
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2



Depth-First Search / Traversal

myGraph.DFS(vertex1)



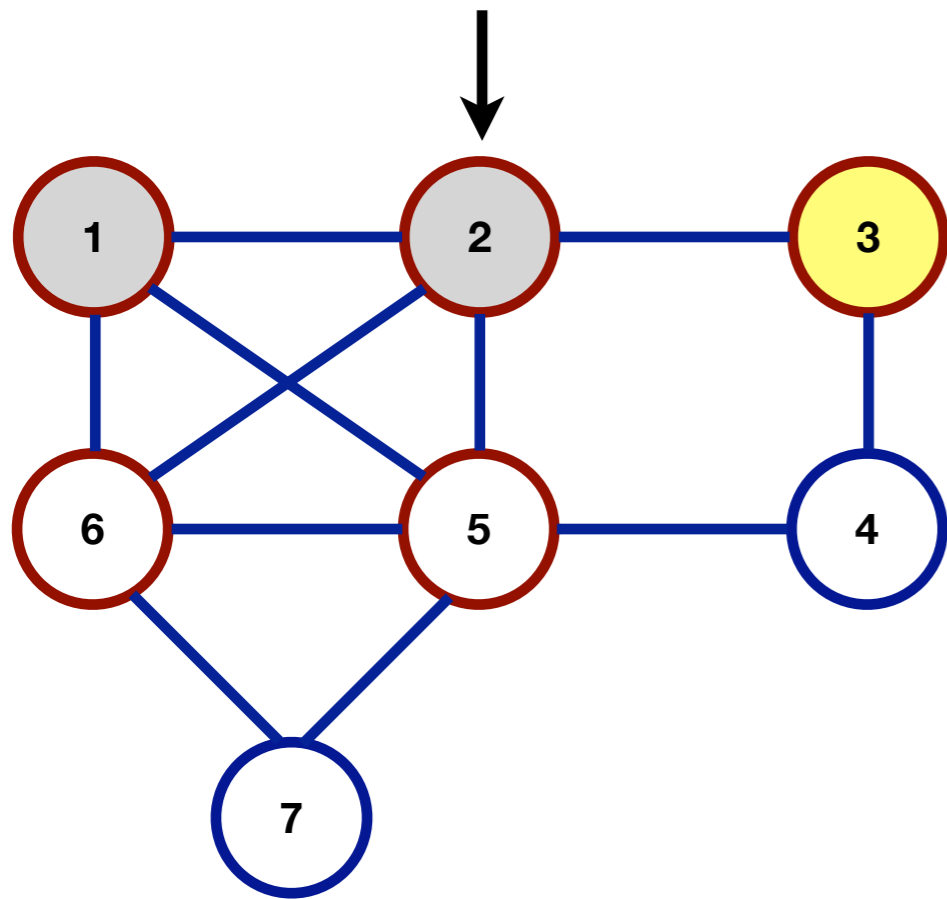
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



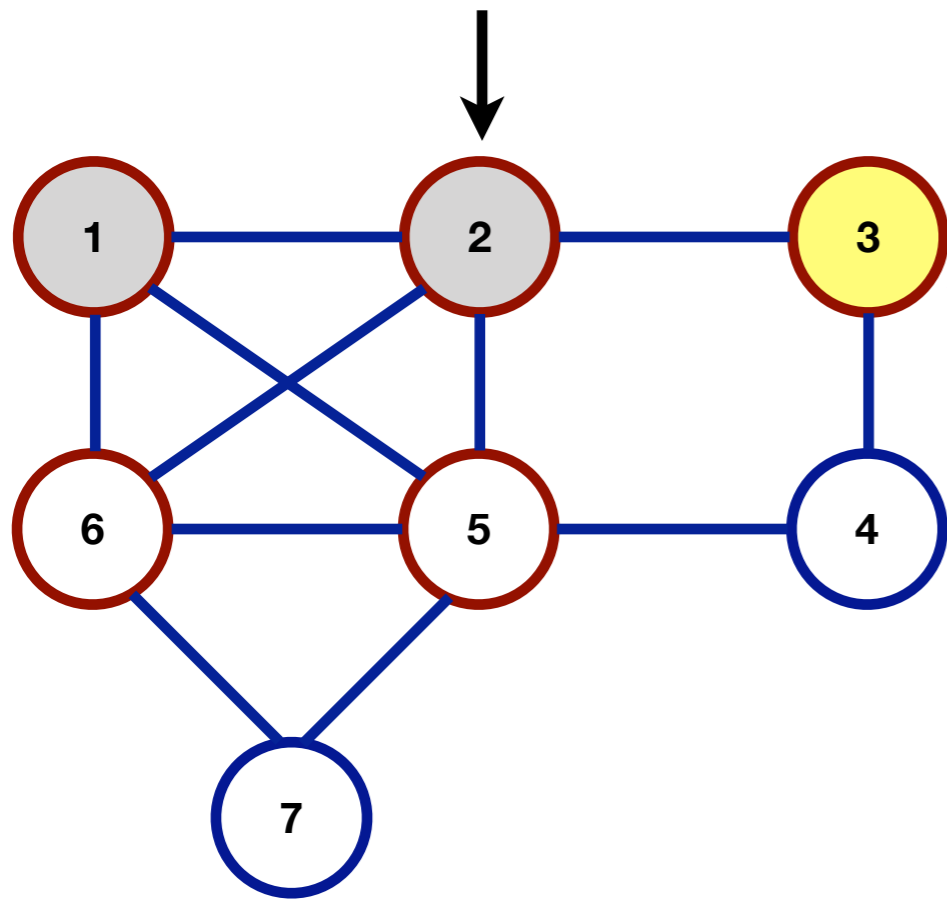
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2

return address
Runtime Stack

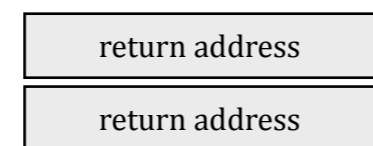
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2

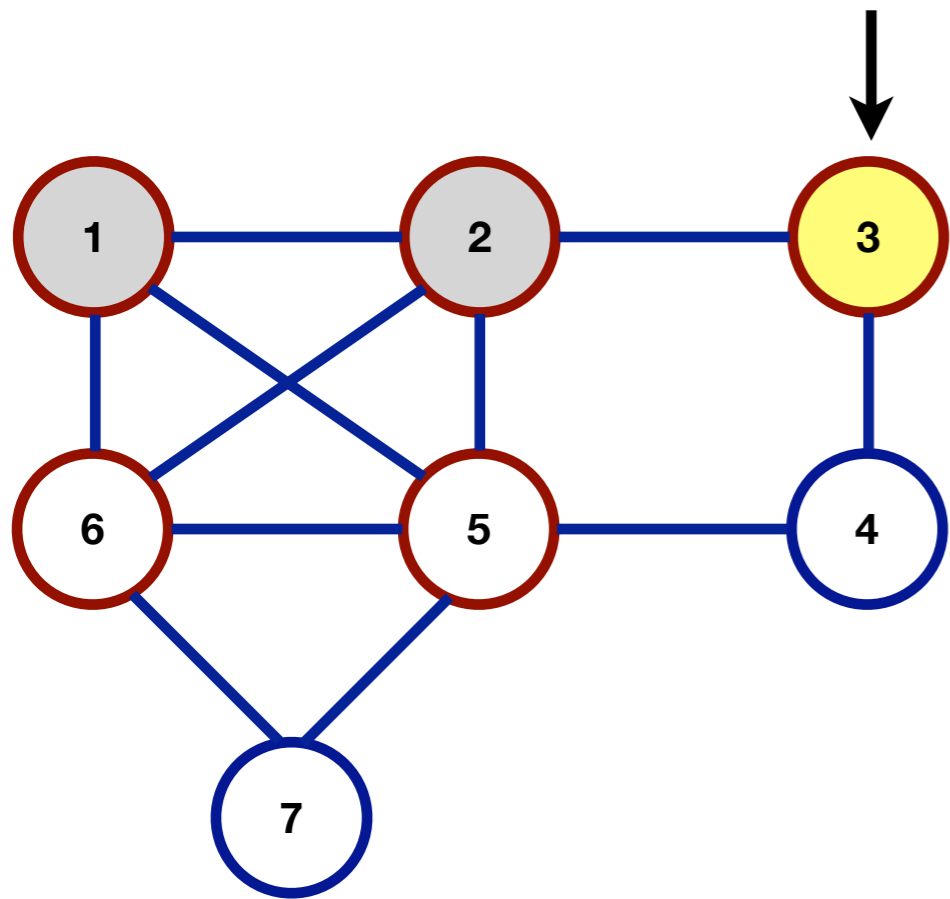


Runtime Stack

Recursive call.
Push return address
on the runtime stack.

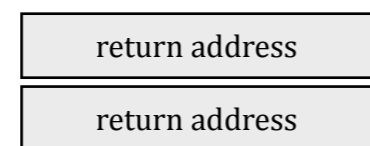
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

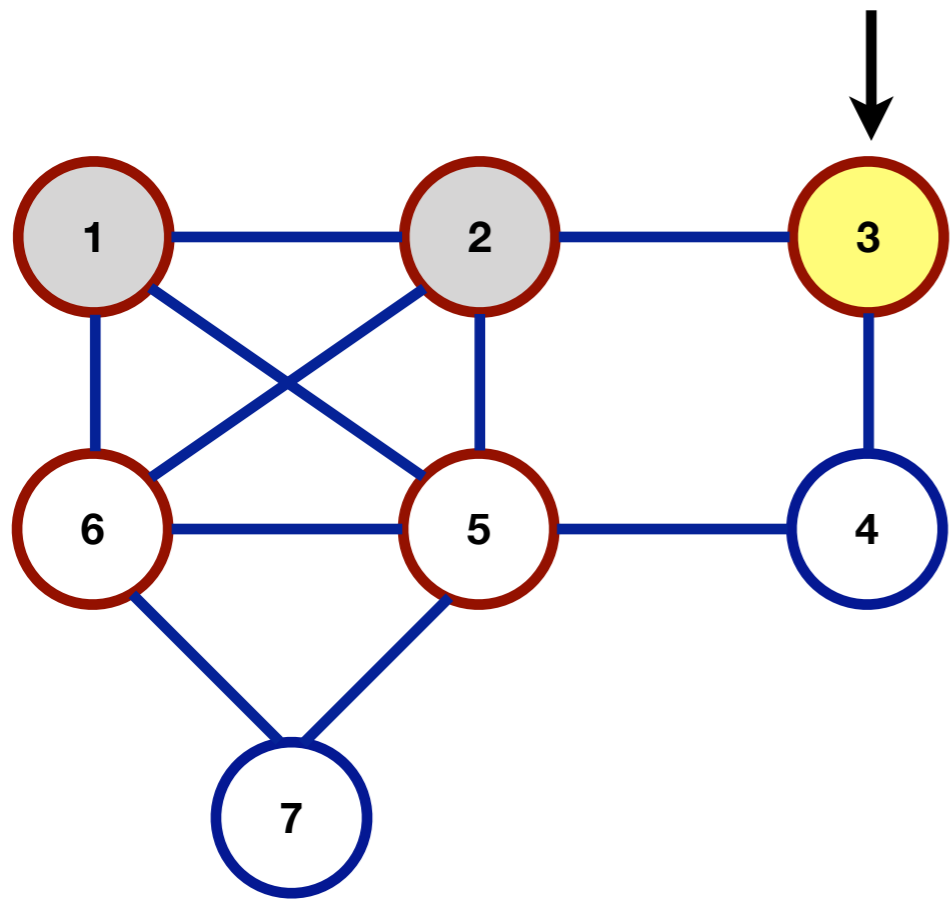
1 2



Runtime Stack

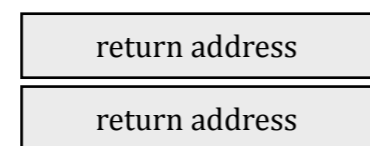
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

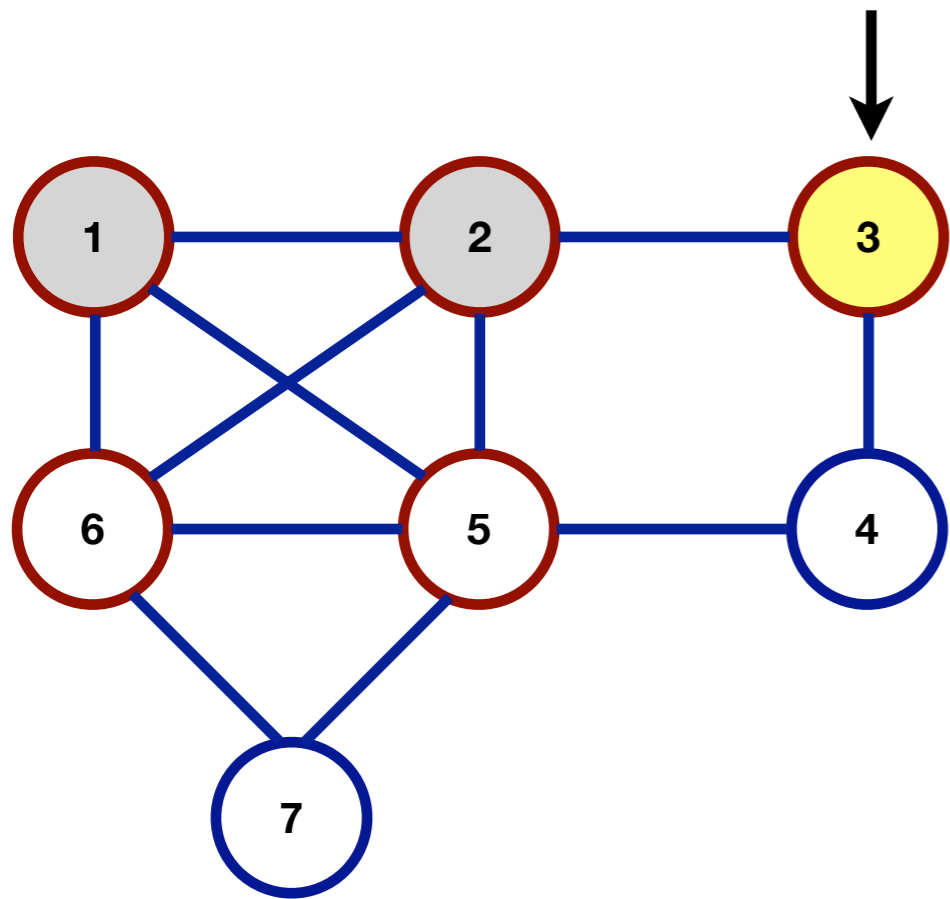
1 2



Runtime Stack

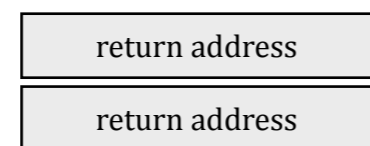
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

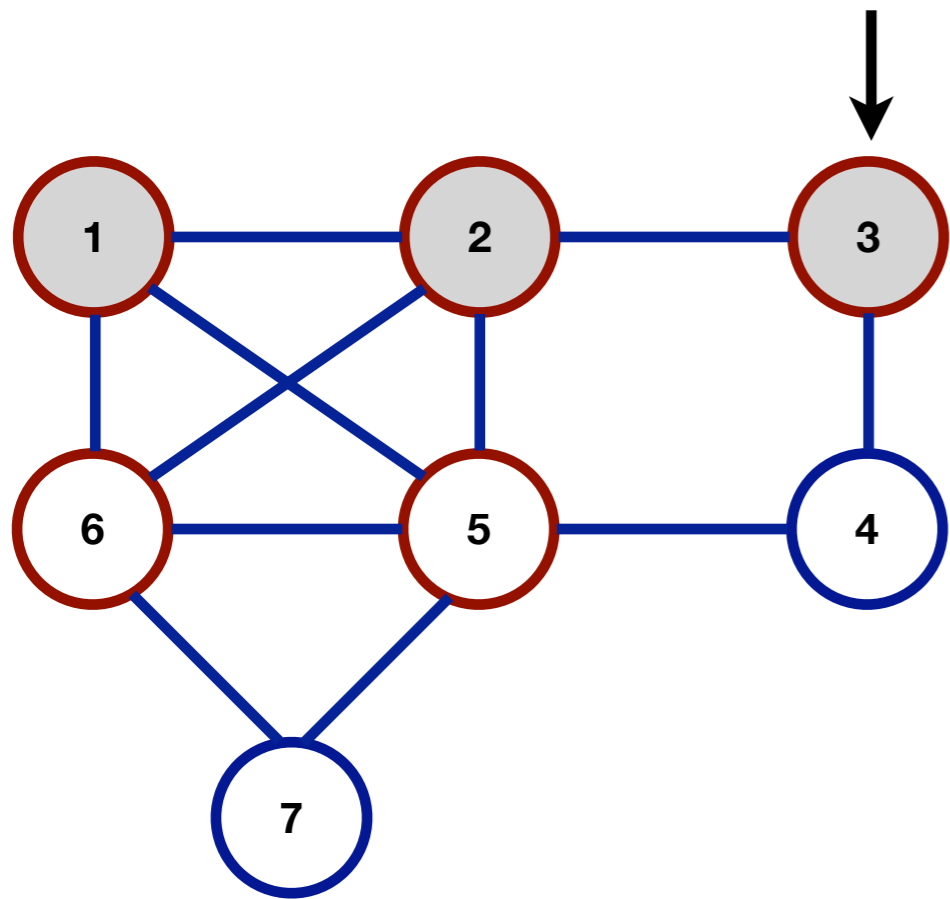
1 2 3



Runtime Stack

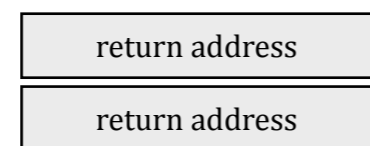
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

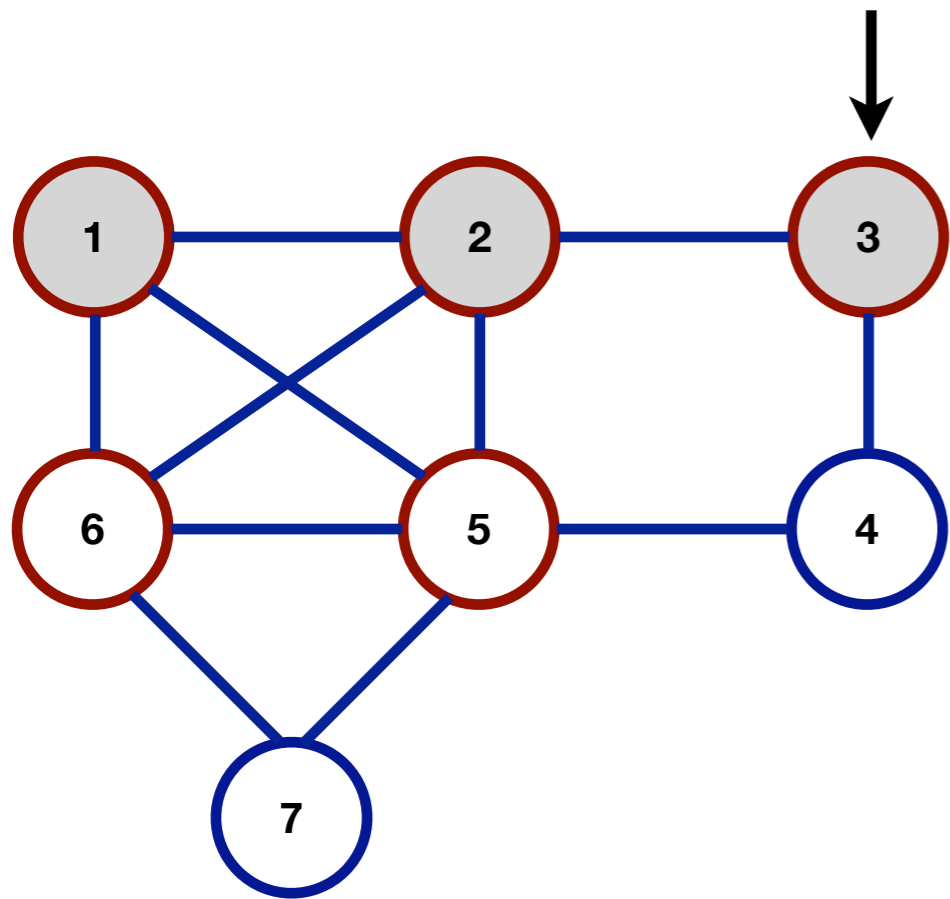
1 2 3



Runtime Stack

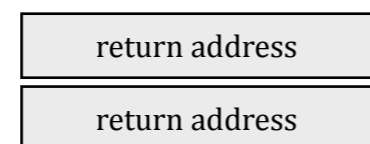
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

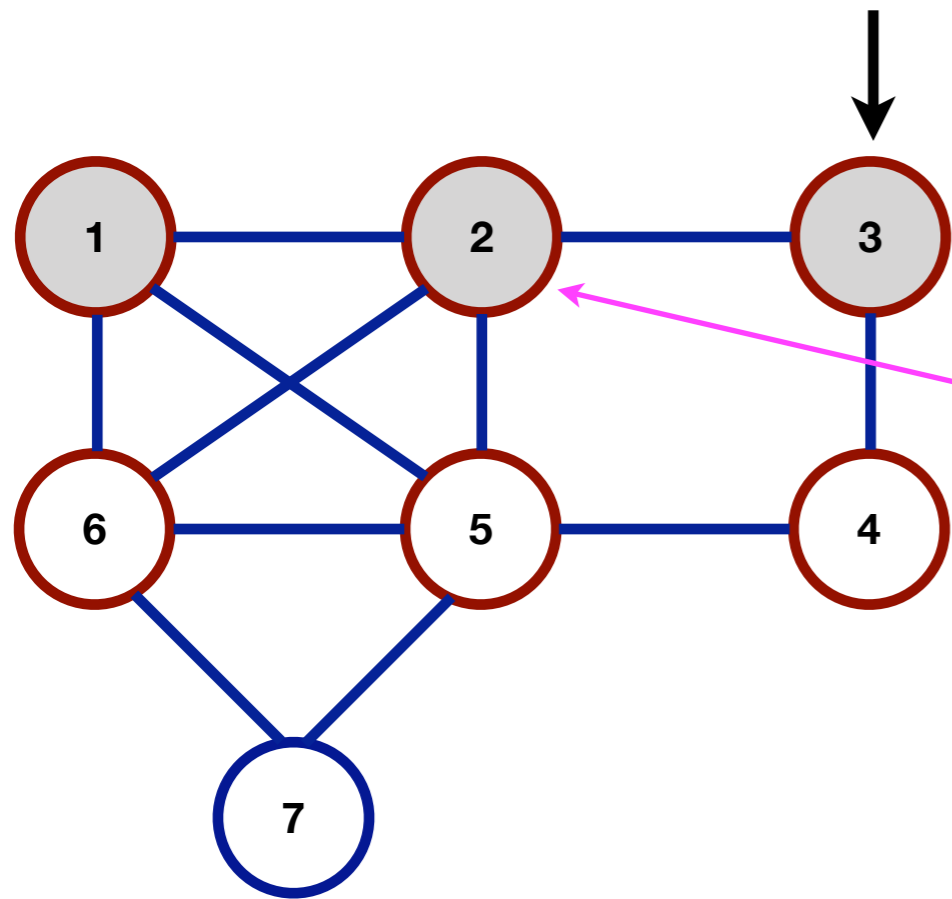
1 2 3



Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)

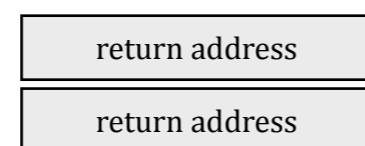


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

```
[1] 2 5 6
[2] 1 3 5 6
[3] 2 4
[4] 3 5
[5] 1 2 4 6 7
[6] 1 2 5 7
[7] 5 6
```

Adjacency List

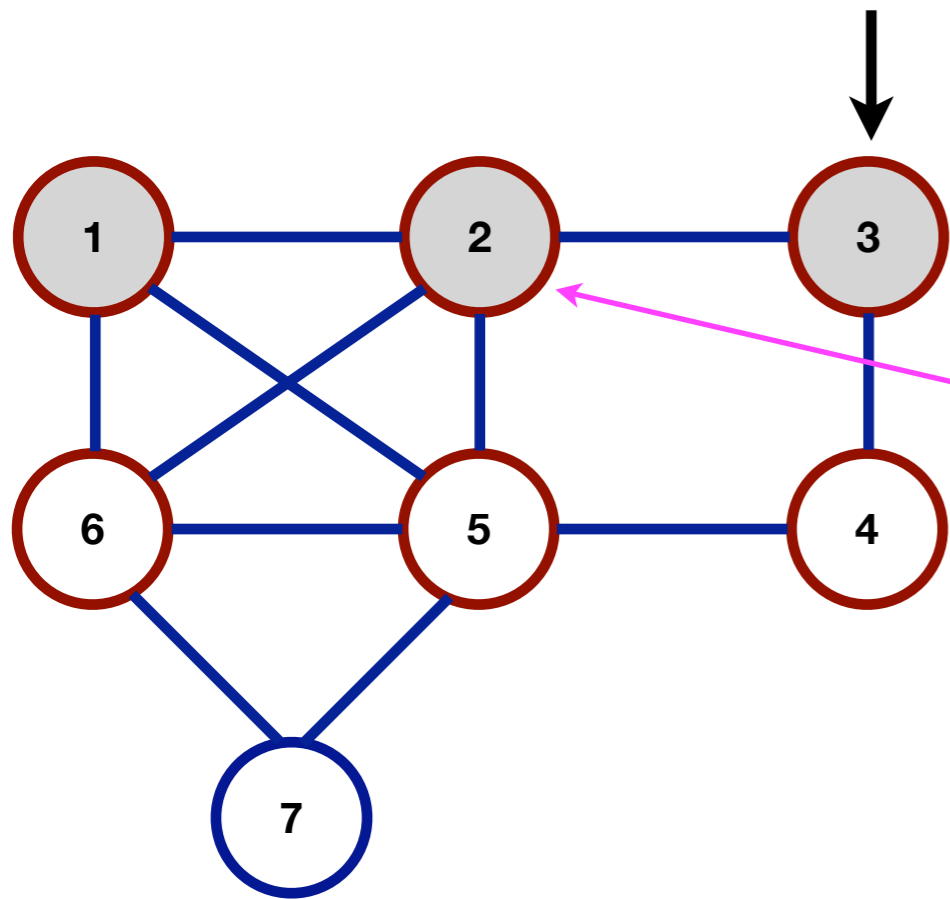
1 2 3



Runtime Stack

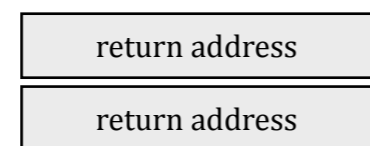
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

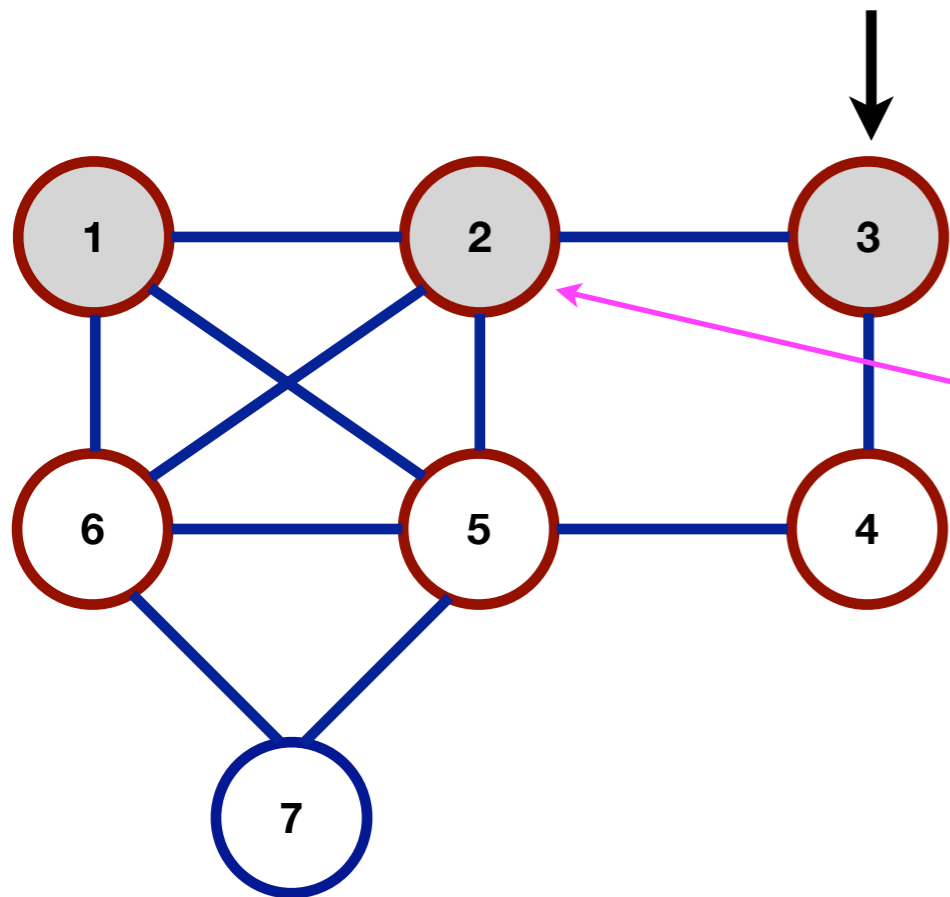
1 2 3



Runtime Stack

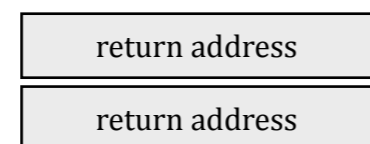
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

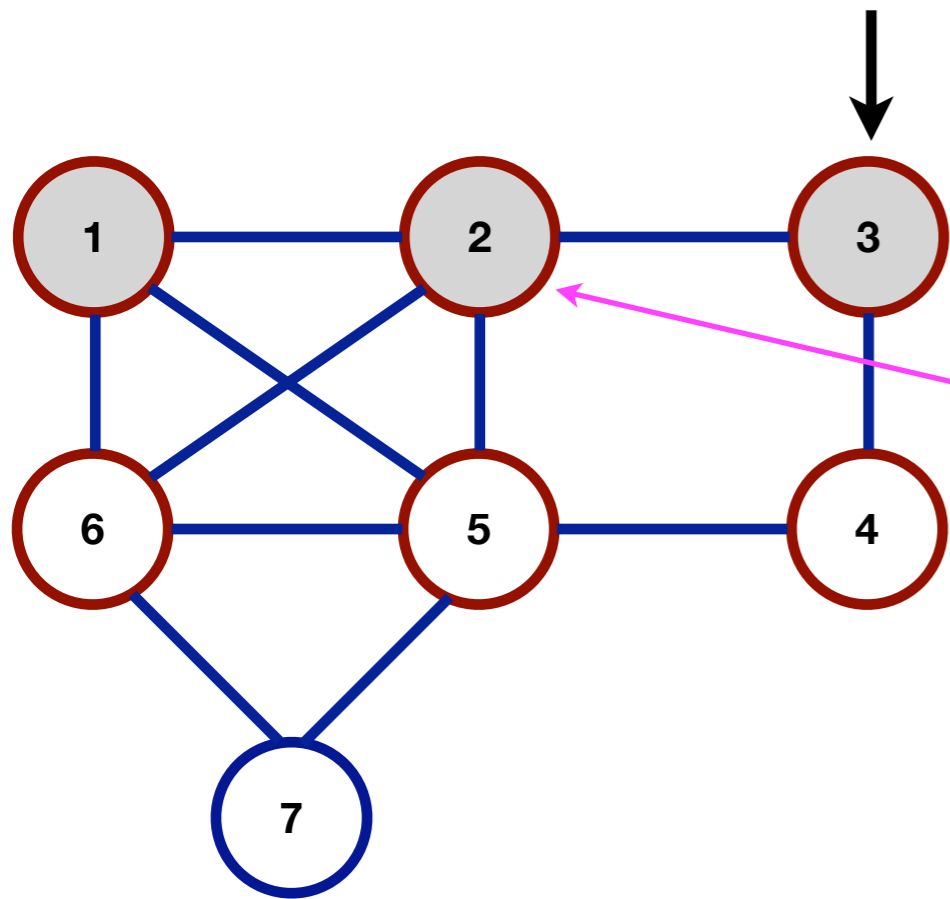
1 2 3



Runtime Stack

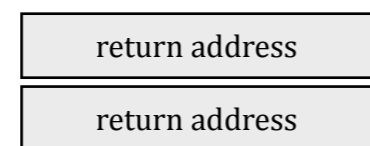
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

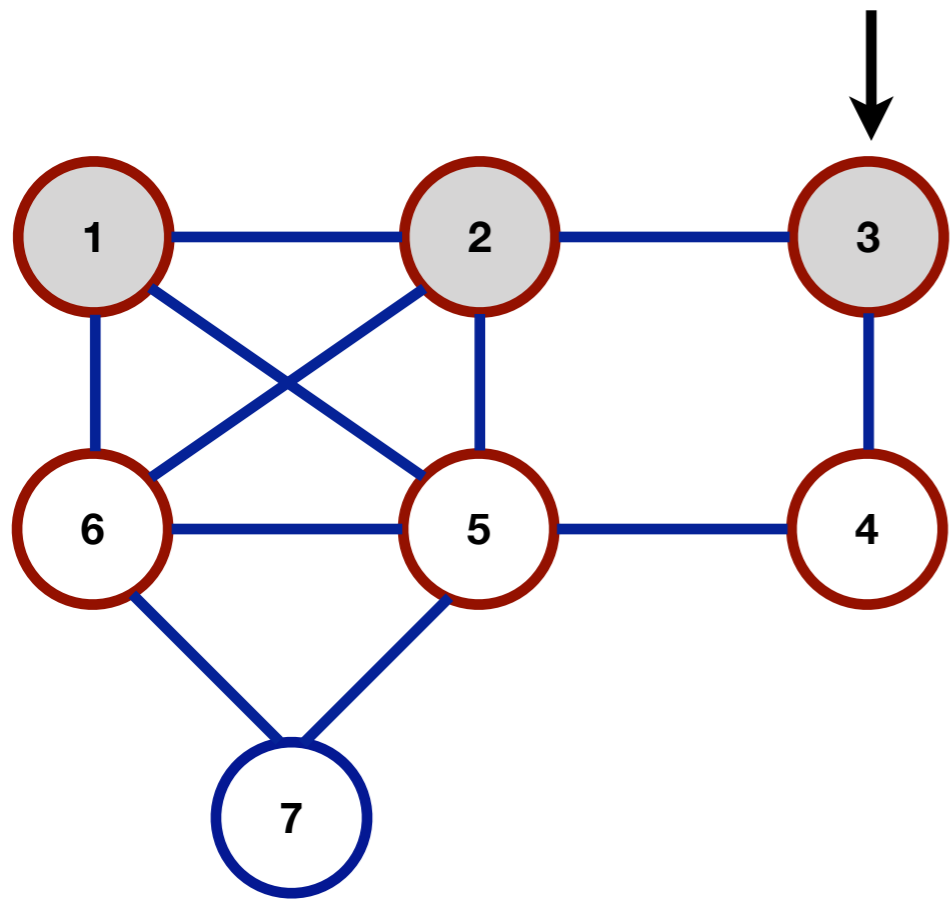
1 2 3



Runtime Stack

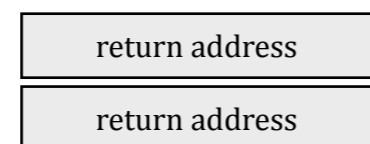
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

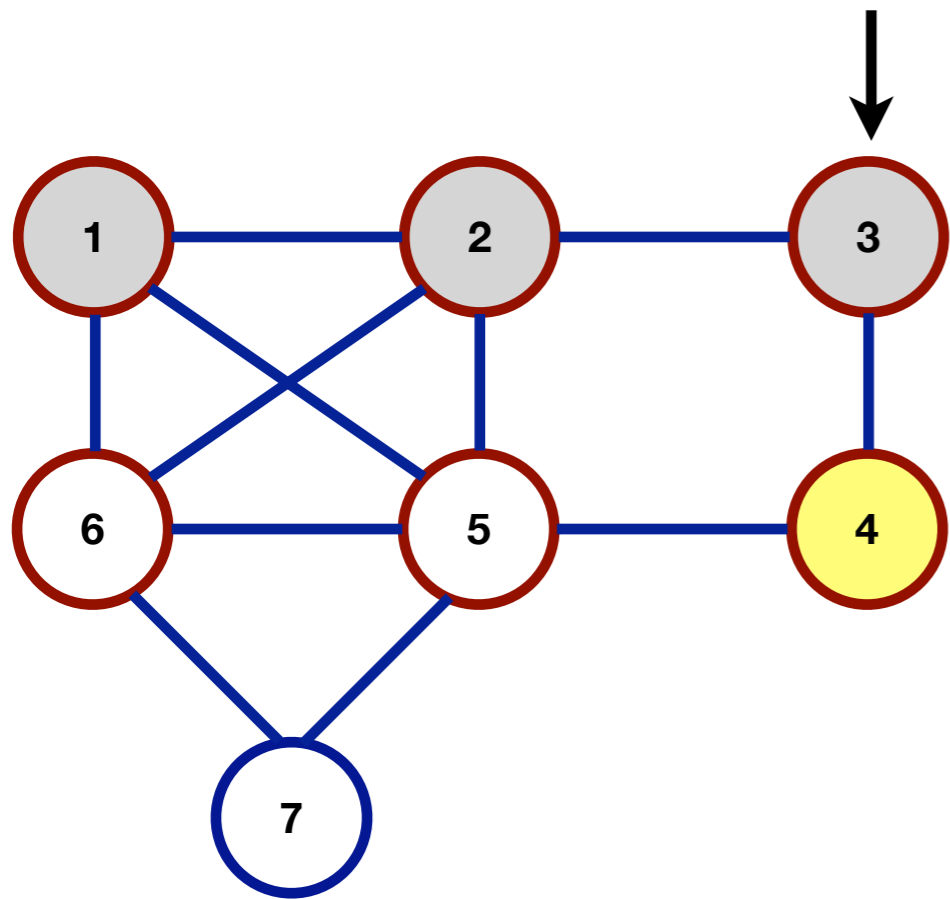
1 2 3



Runtime Stack

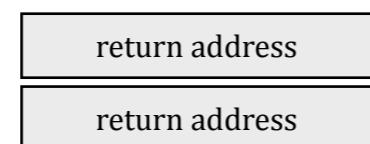
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

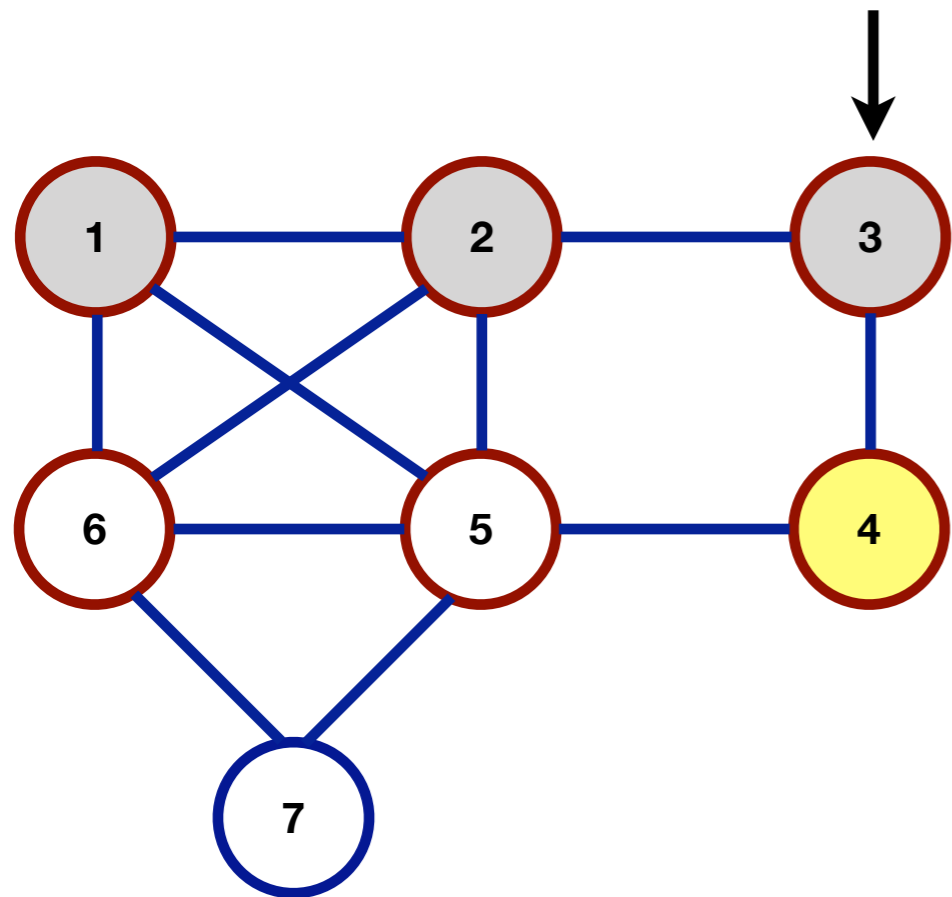
1 2 3



Runtime Stack

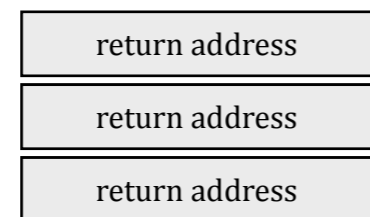
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

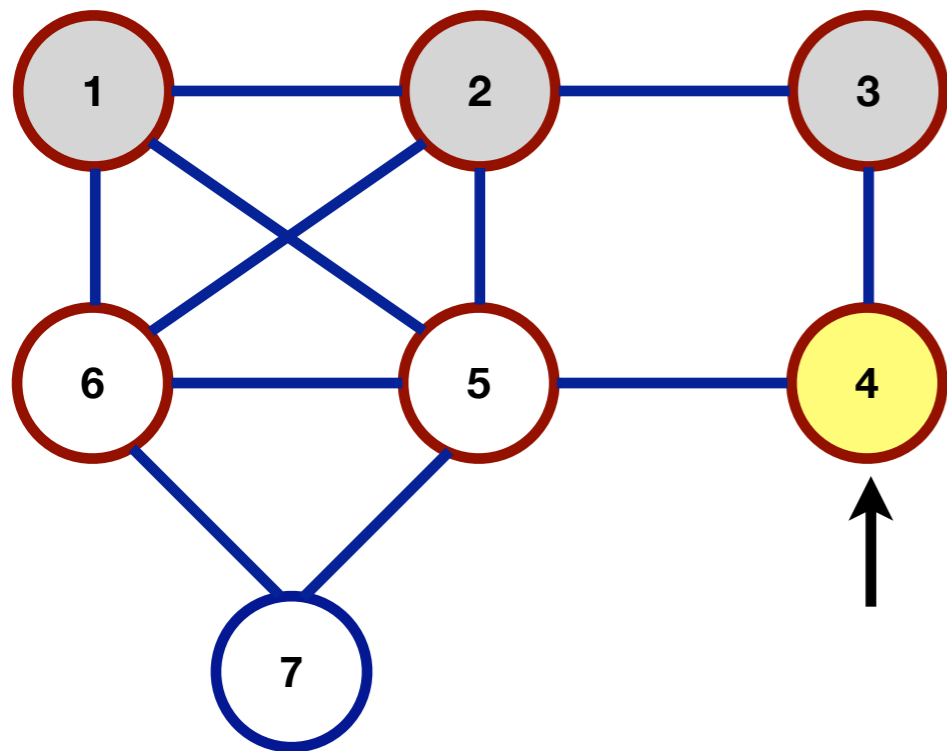
1 2 3



Runtime Stack

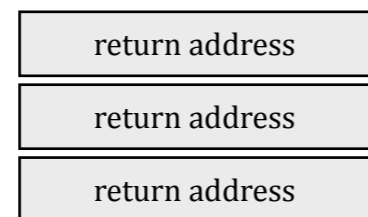
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

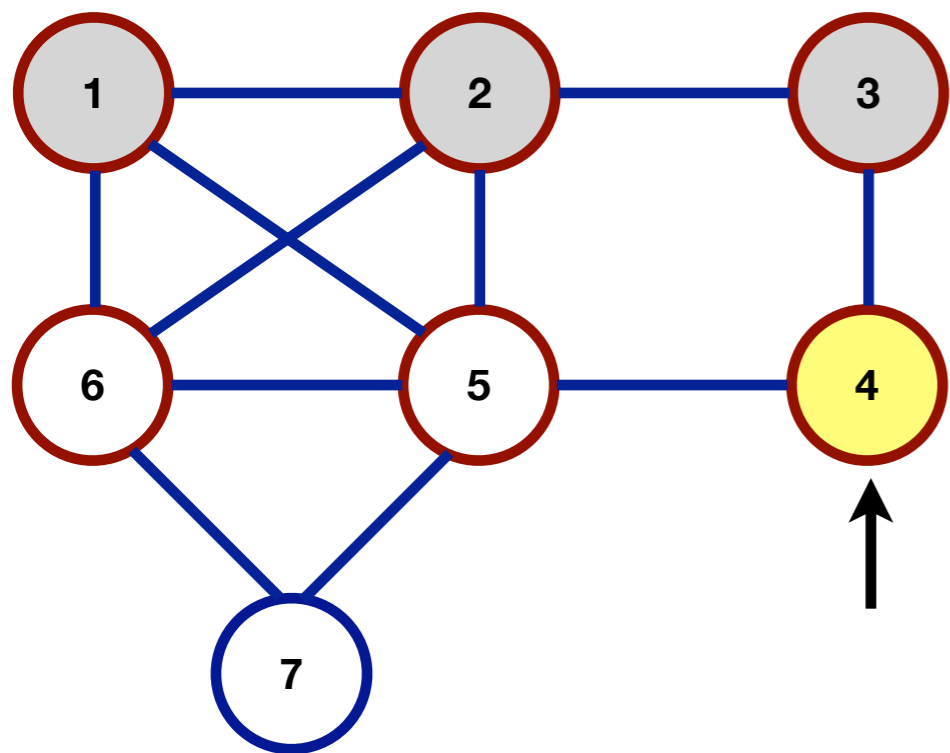
1 2 3



Runtime Stack

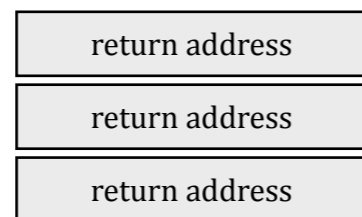
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

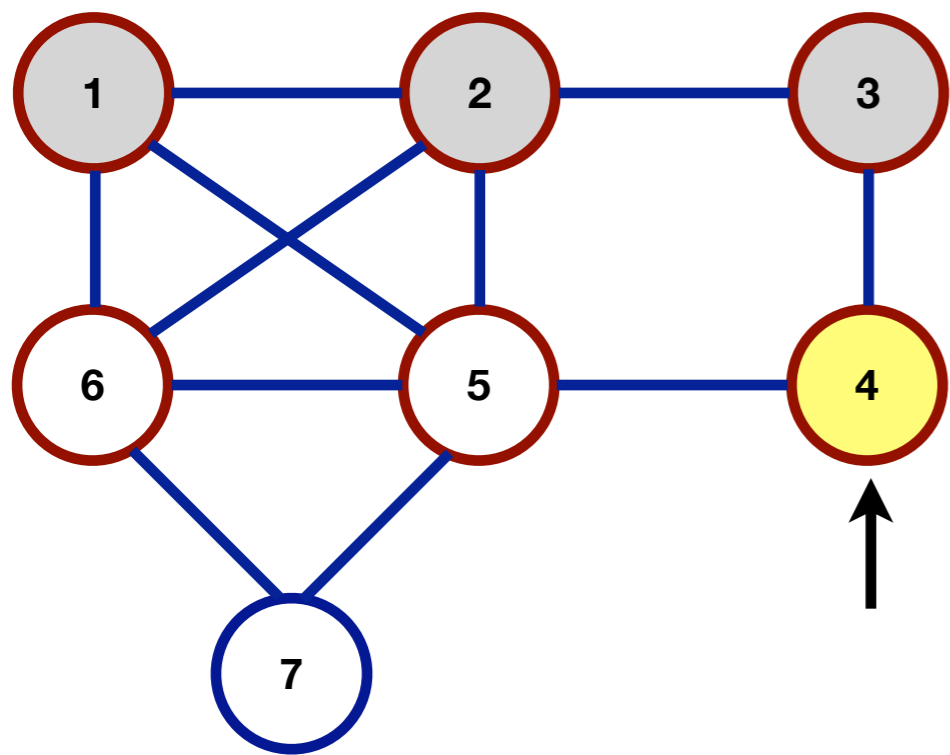
1 2 3



Runtime Stack

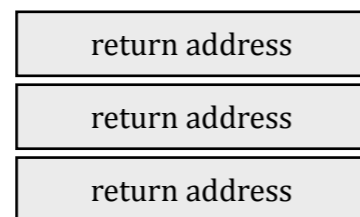
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

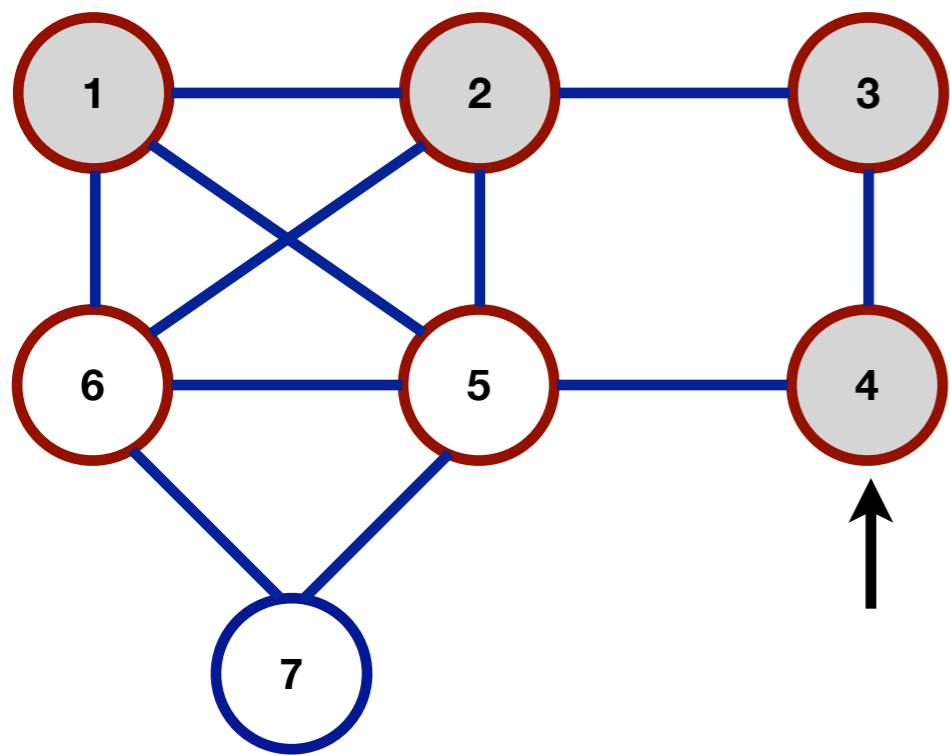
1 2 3 4



Runtime Stack

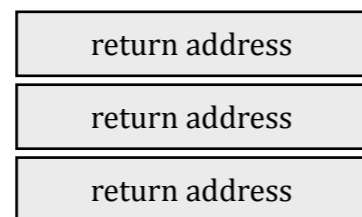
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

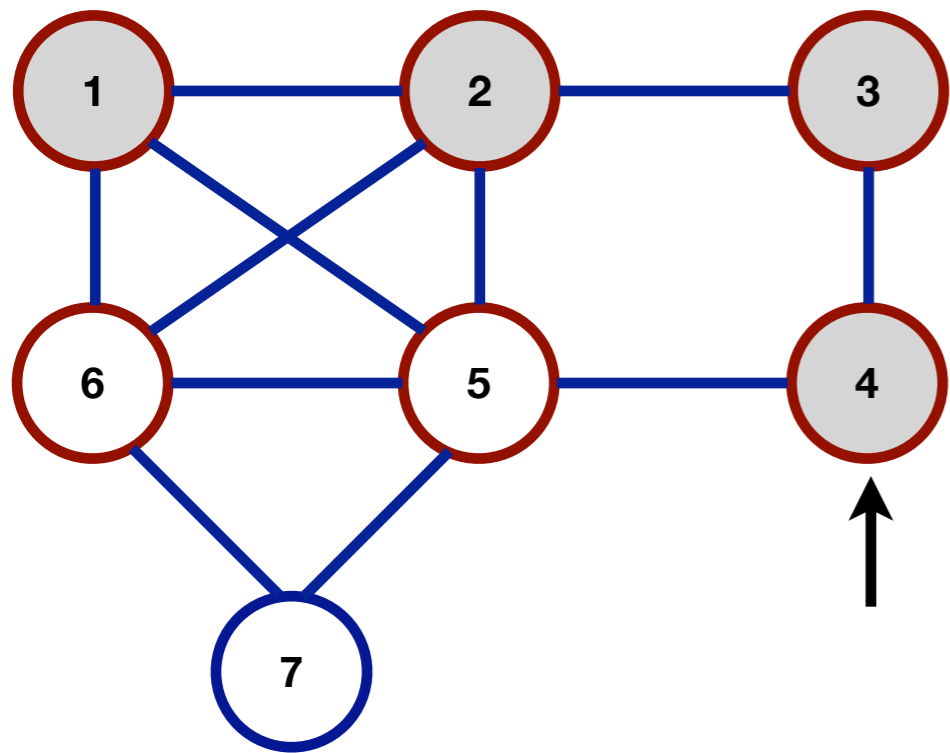
1 2 3 4



Runtime Stack

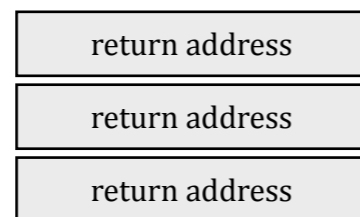
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

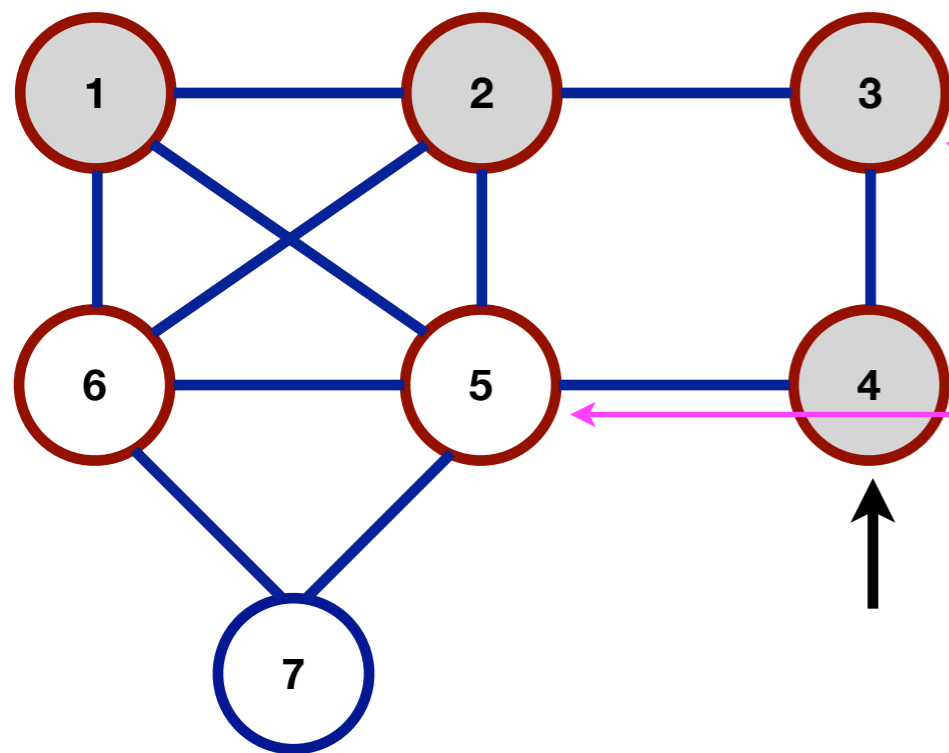
1 2 3 4



Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)

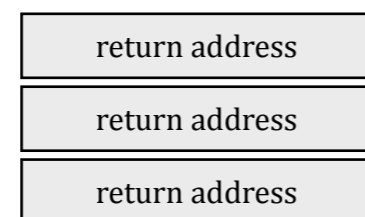


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

```
[1] 2 5 6
[2] 1 3 5 6
[3] 2 4
[4] 3 5
[5] 1 2 4 6 7
[6] 1 2 5 7
[7] 5 6
```

Adjacency List

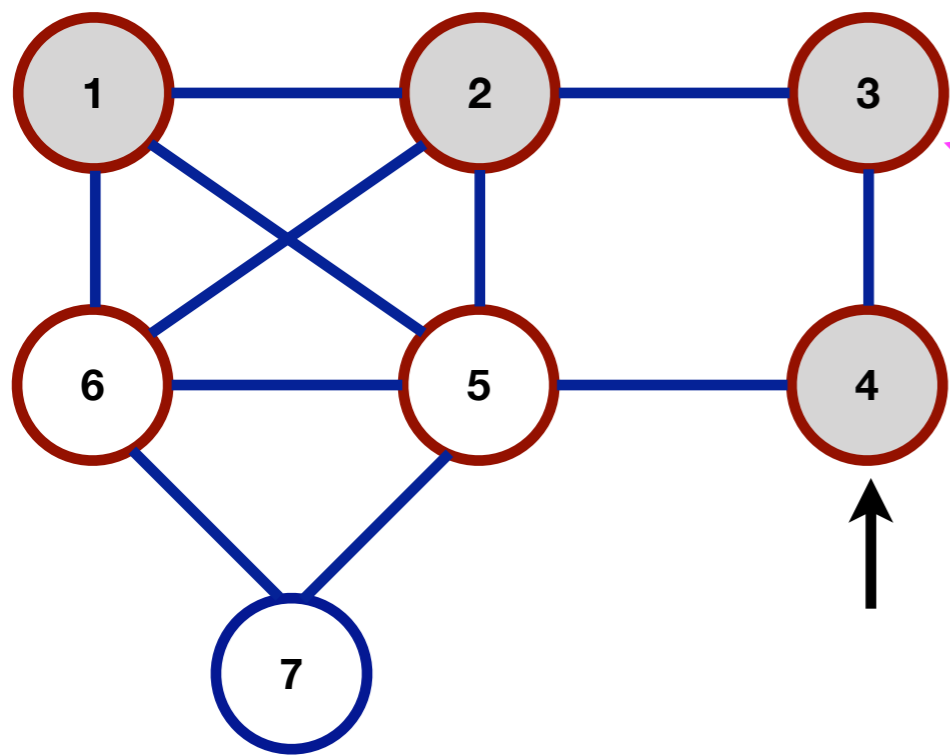
1 2 3 4



Runtime Stack

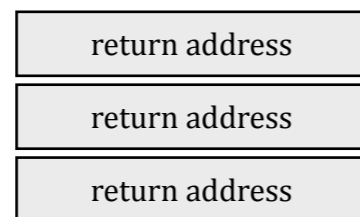
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

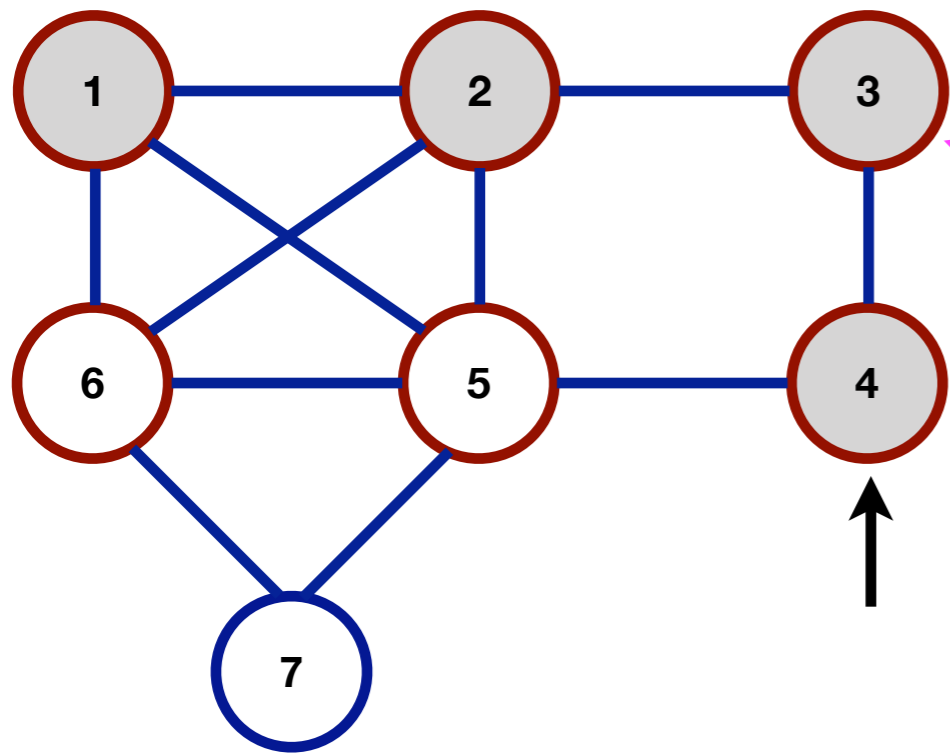
1 2 3 4



Runtime Stack

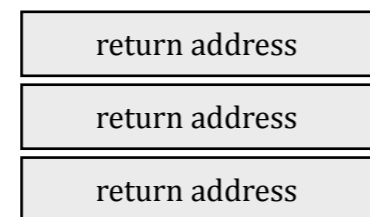
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

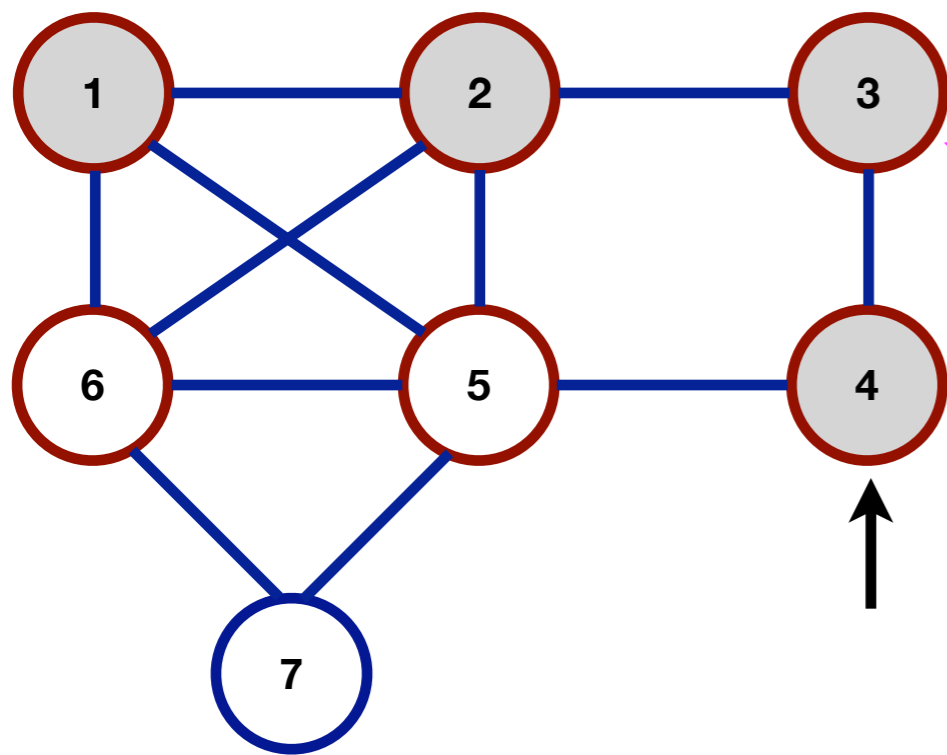
1 2 3 4



Runtime Stack

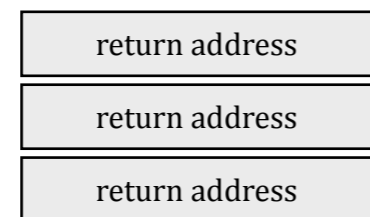
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

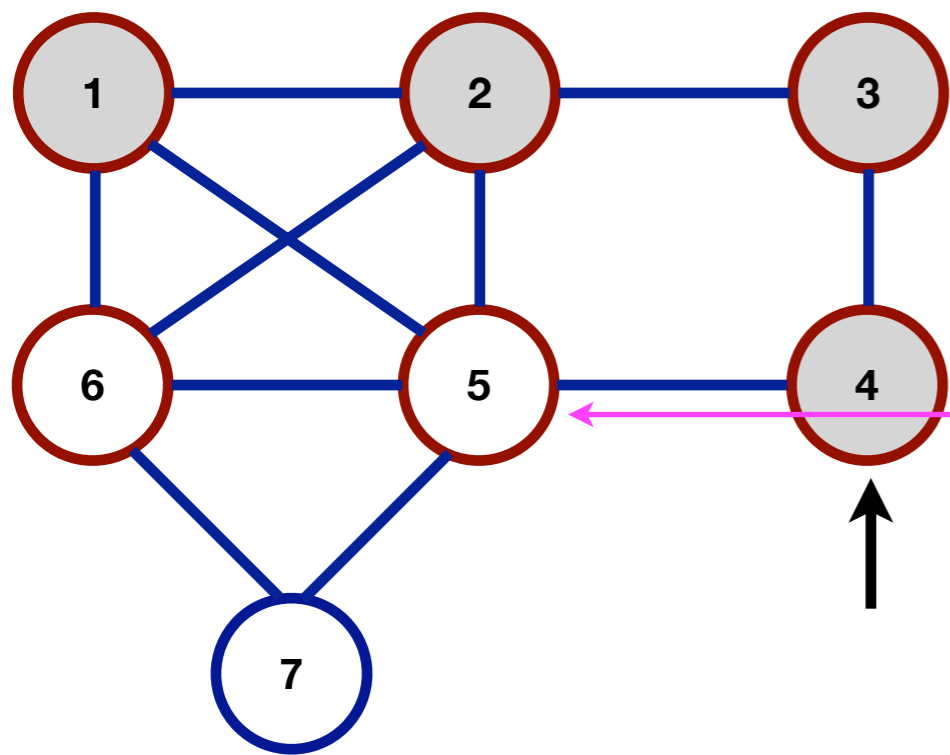
1 2 3 4



Runtime Stack

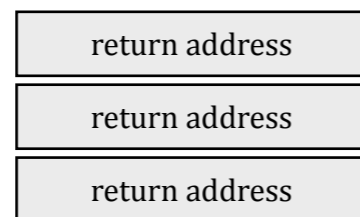
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

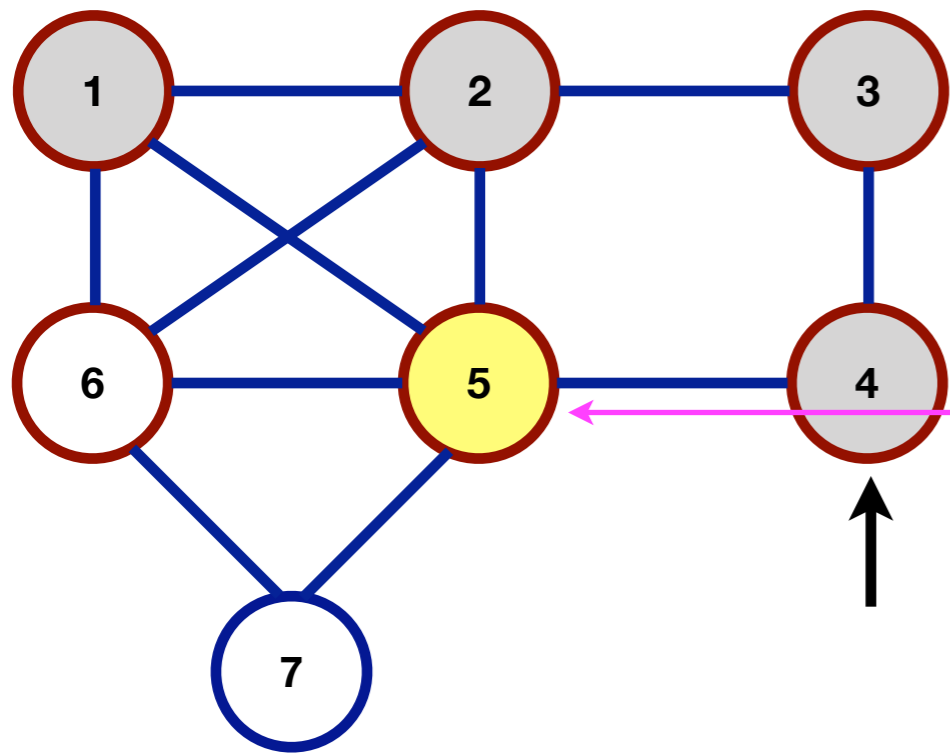
1 2 3 4



Runtime Stack

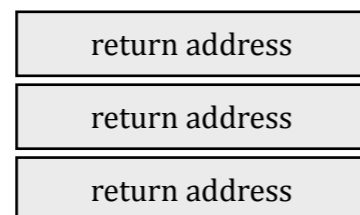
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

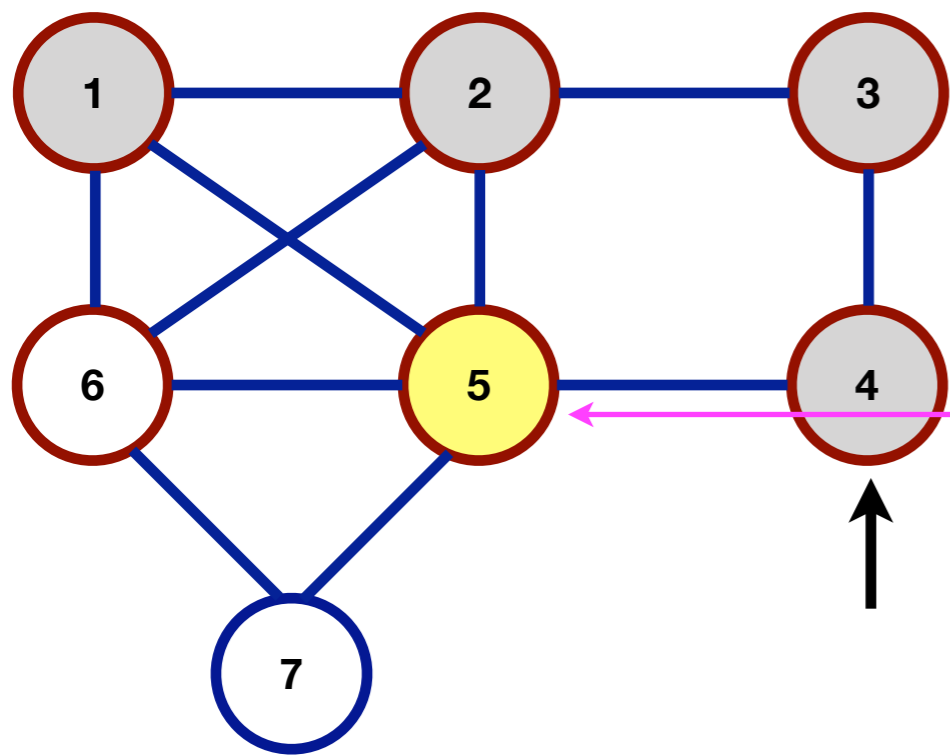
1 2 3 4



Runtime Stack

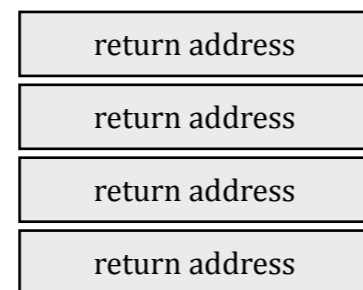
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

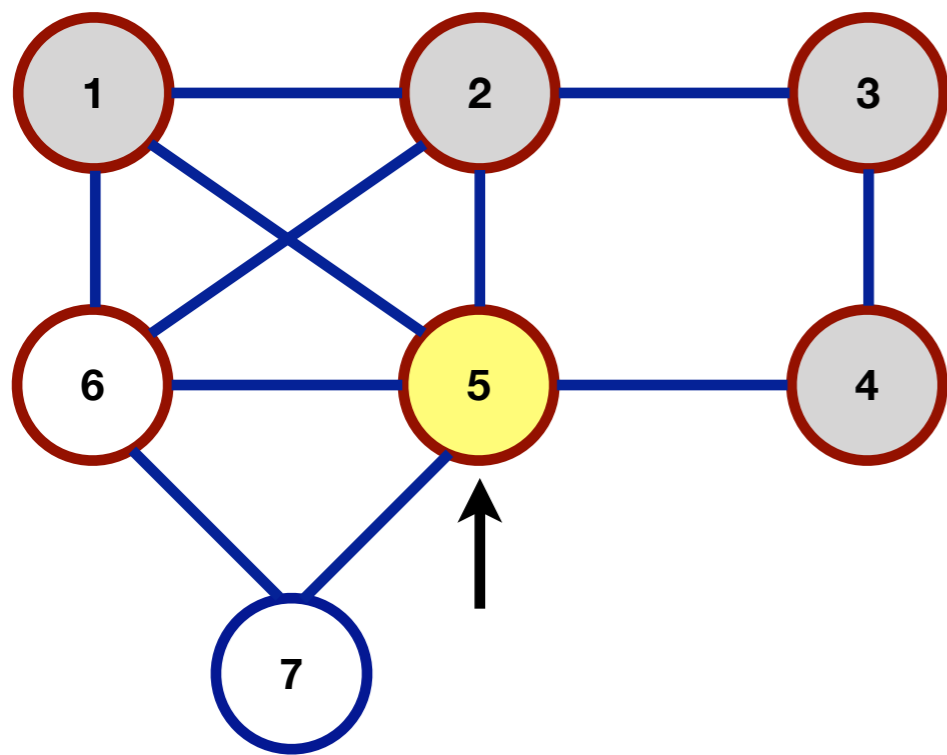
1 2 3 4



Runtime Stack

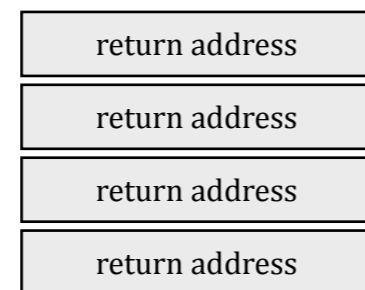
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

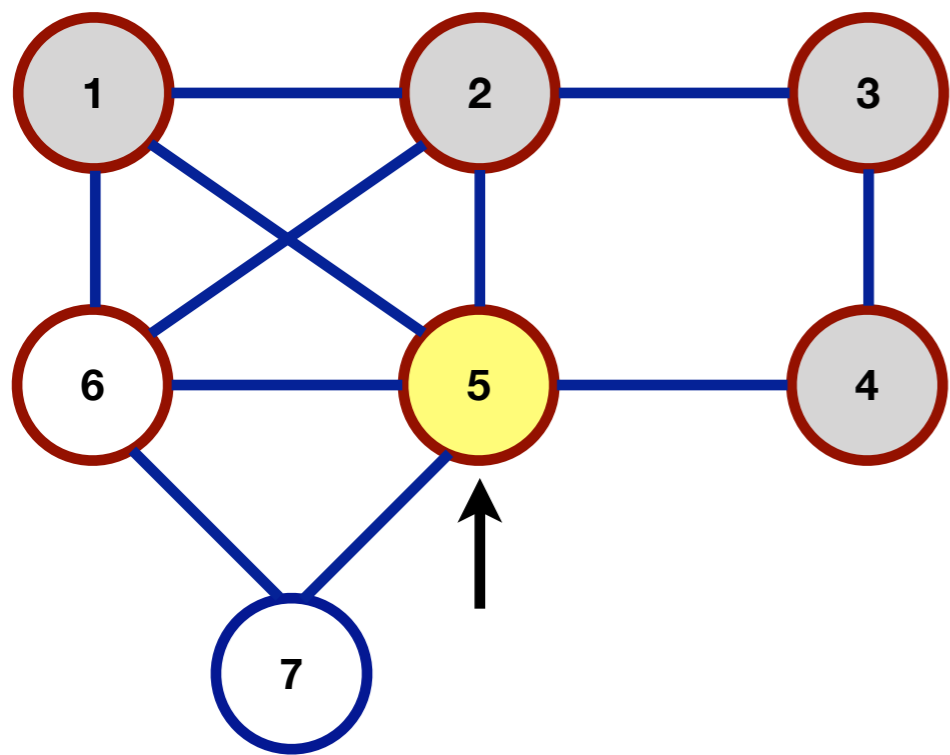
1 2 3 4



Runtime Stack

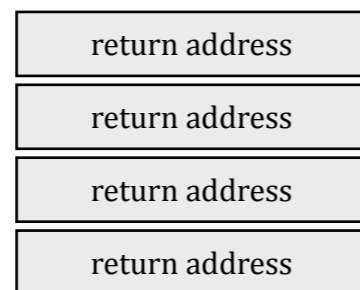
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

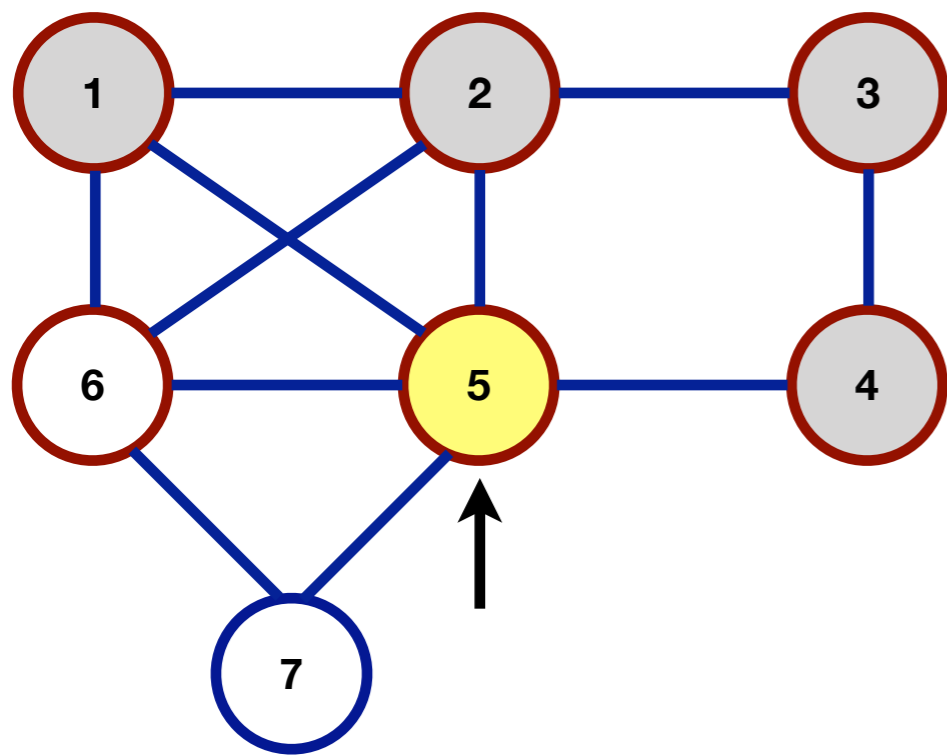
1 2 3 4



Runtime Stack

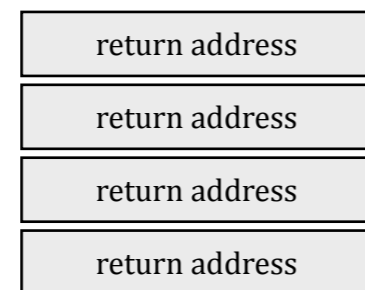
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

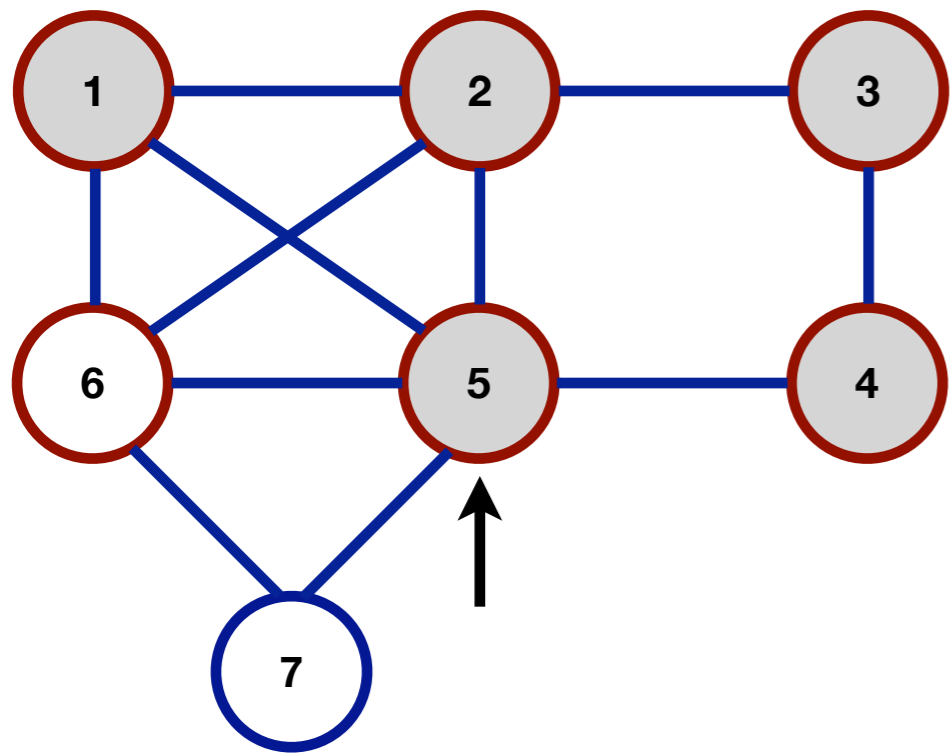
1 2 3 4 5



Runtime Stack

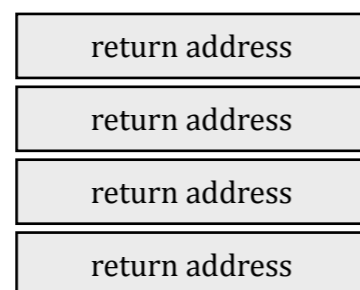
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

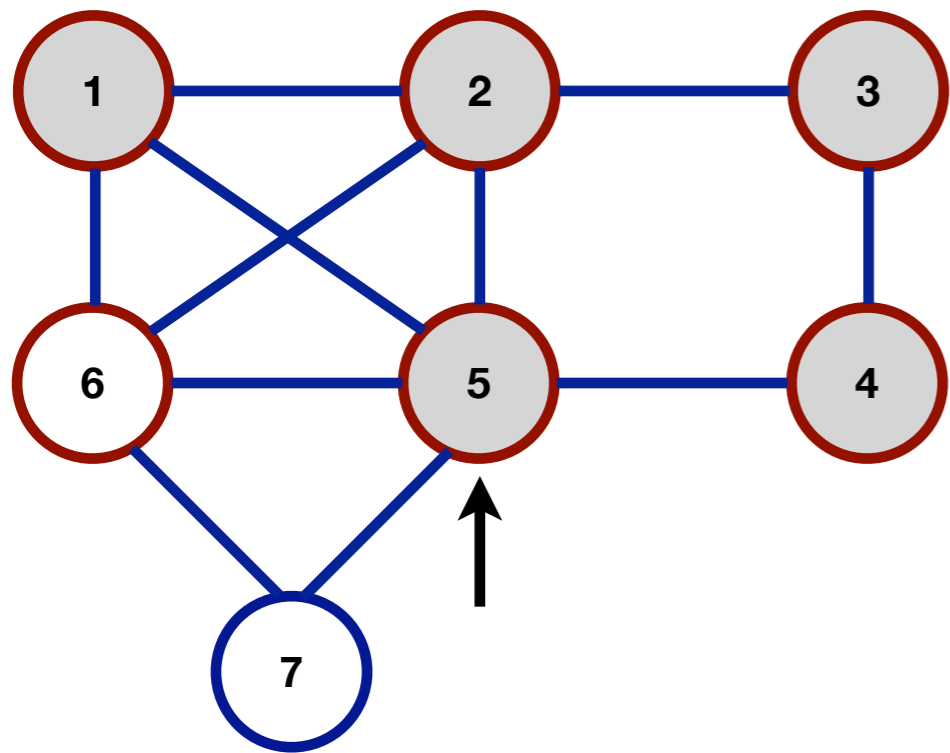
1 2 3 4 5



Runtime Stack

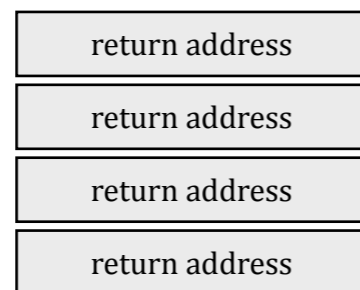
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

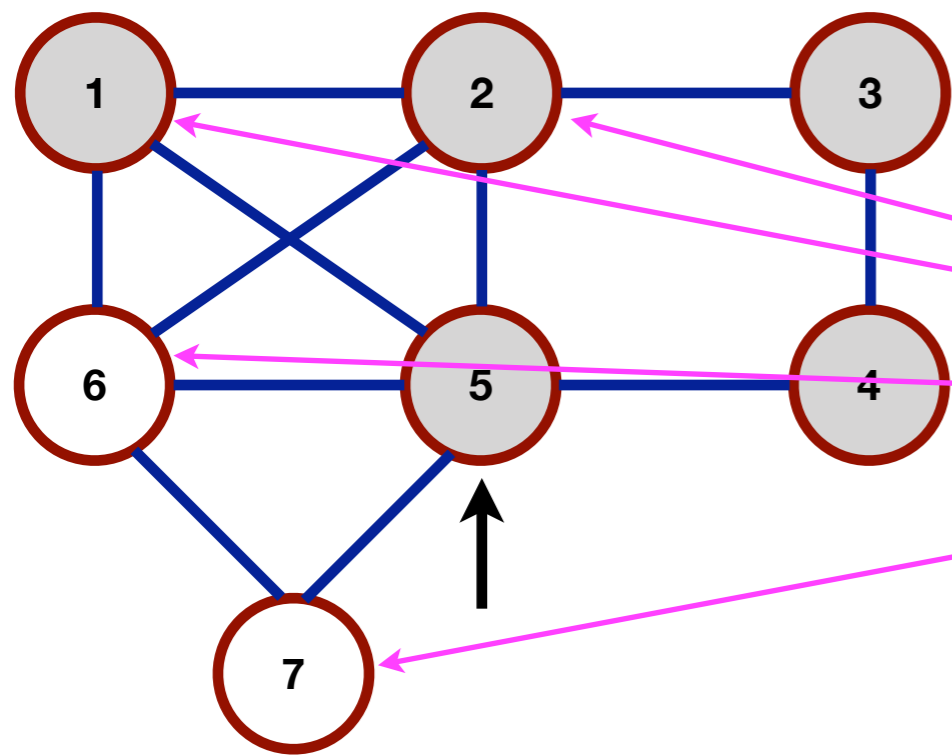
1 2 3 4 5



Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



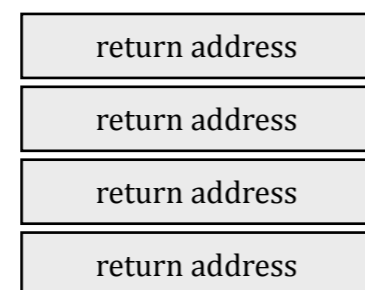
```

proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
    
```

[1]	2	5	6		
[2]	1	3	5	6	
[3]	2	4			
[4]	3	5			
[5]	1	2	4	6	7
[6]	1	2	5	7	
[7]	5	6			

Adjacency List

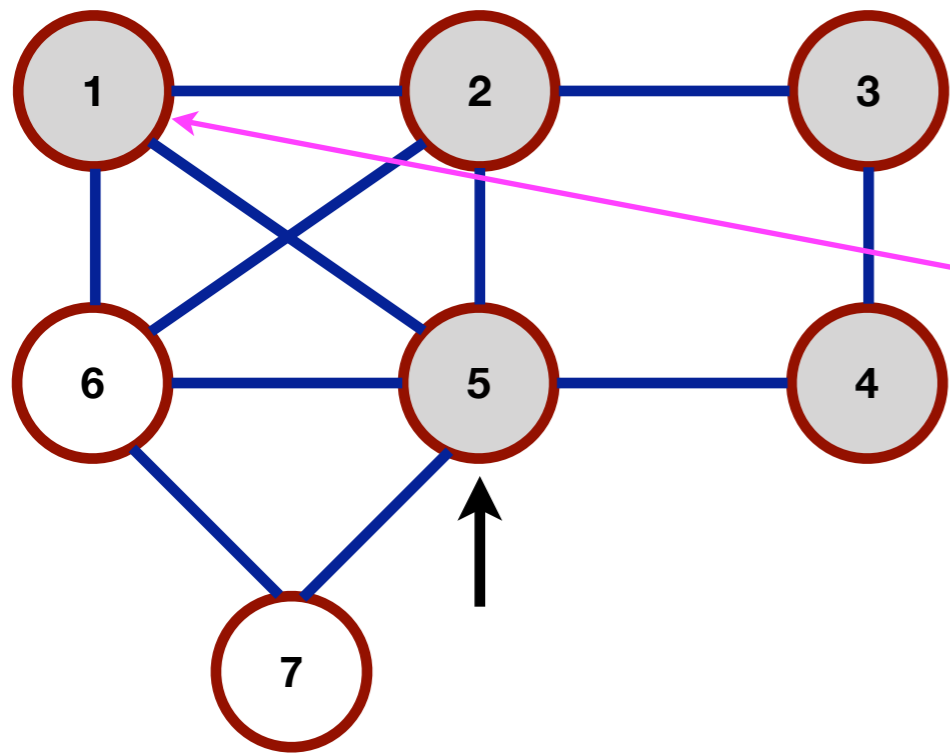
1 2 3 4 5



Runtime Stack

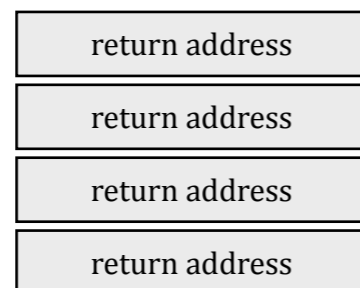
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

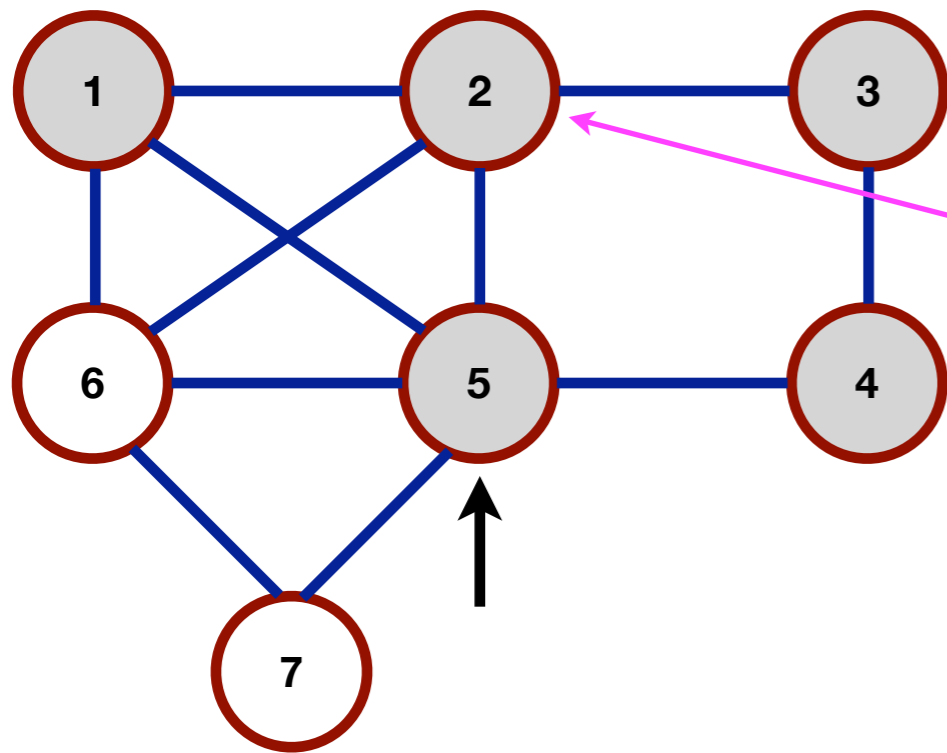
1 2 3 4 5



Runtime Stack

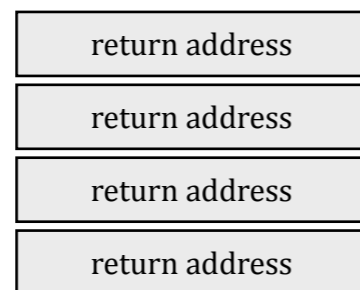
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

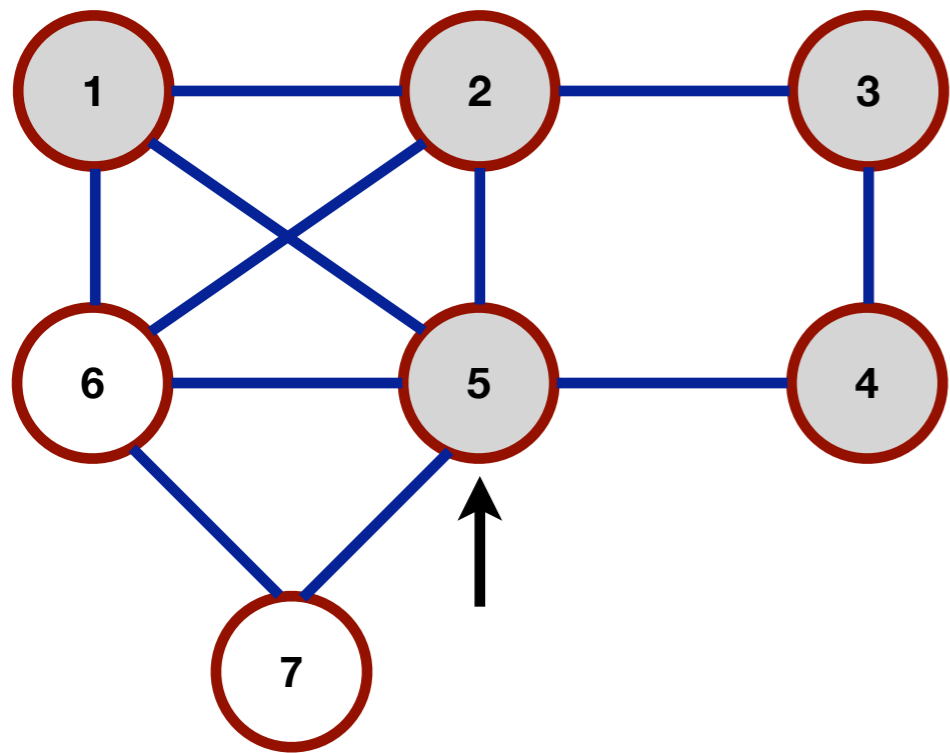
1 2 3 4 5



Runtime Stack

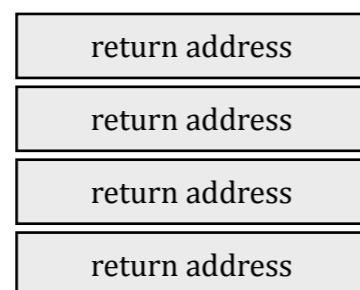
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

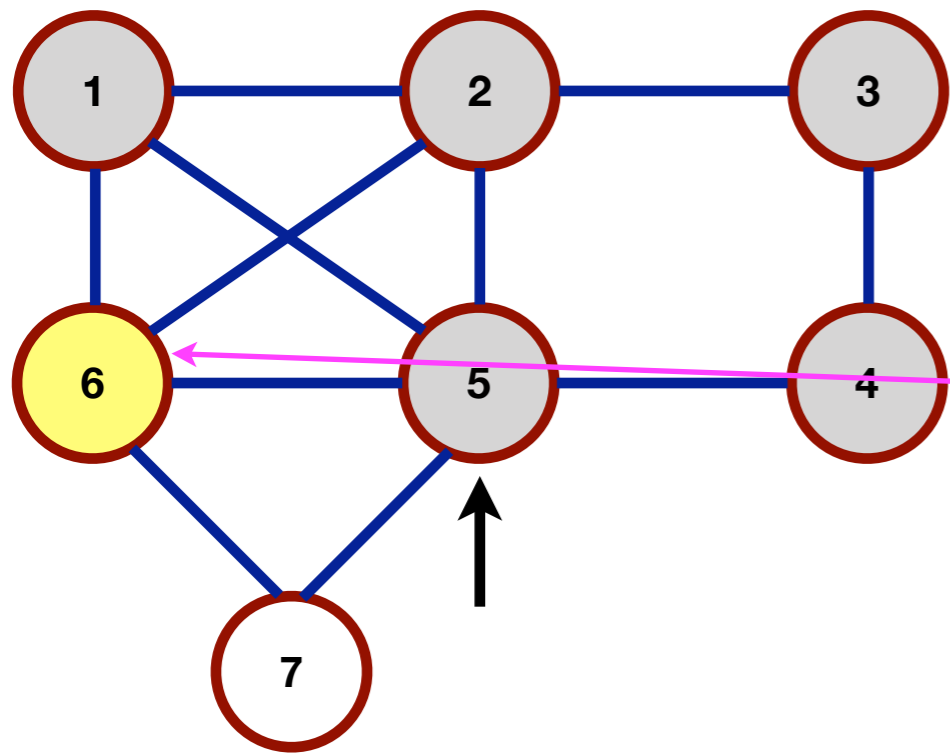
1 2 3 4 5



Runtime Stack

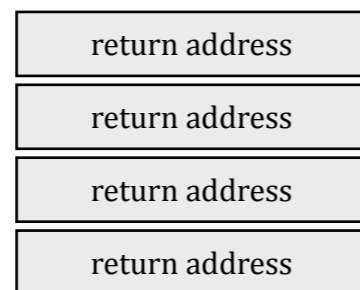
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

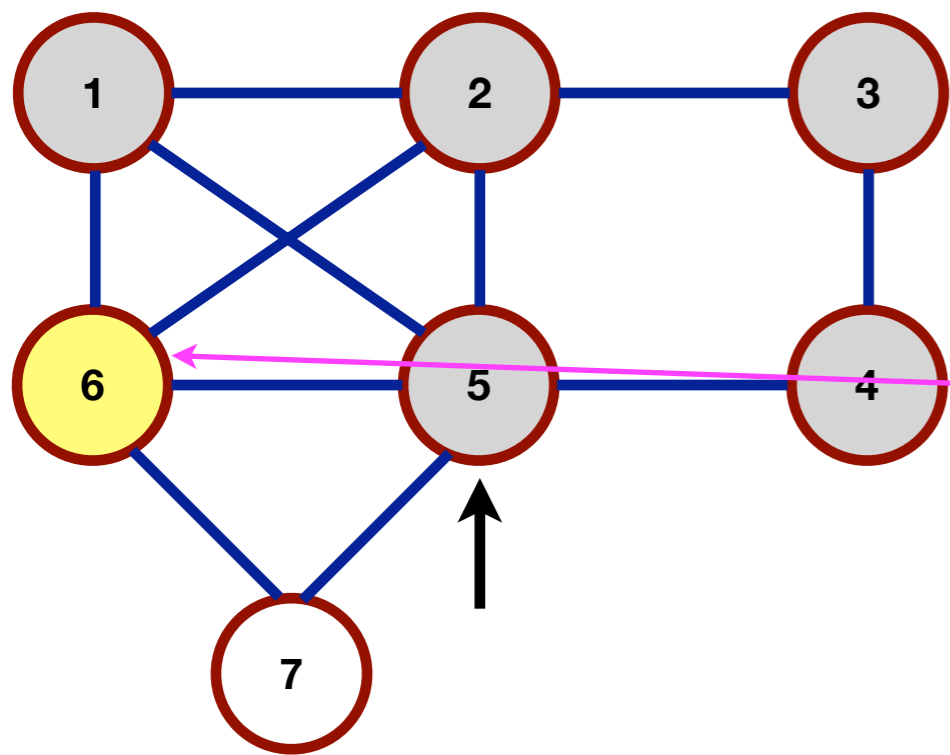
1 2 3 4 5



Runtime Stack

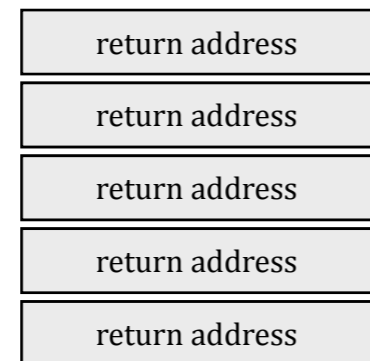
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

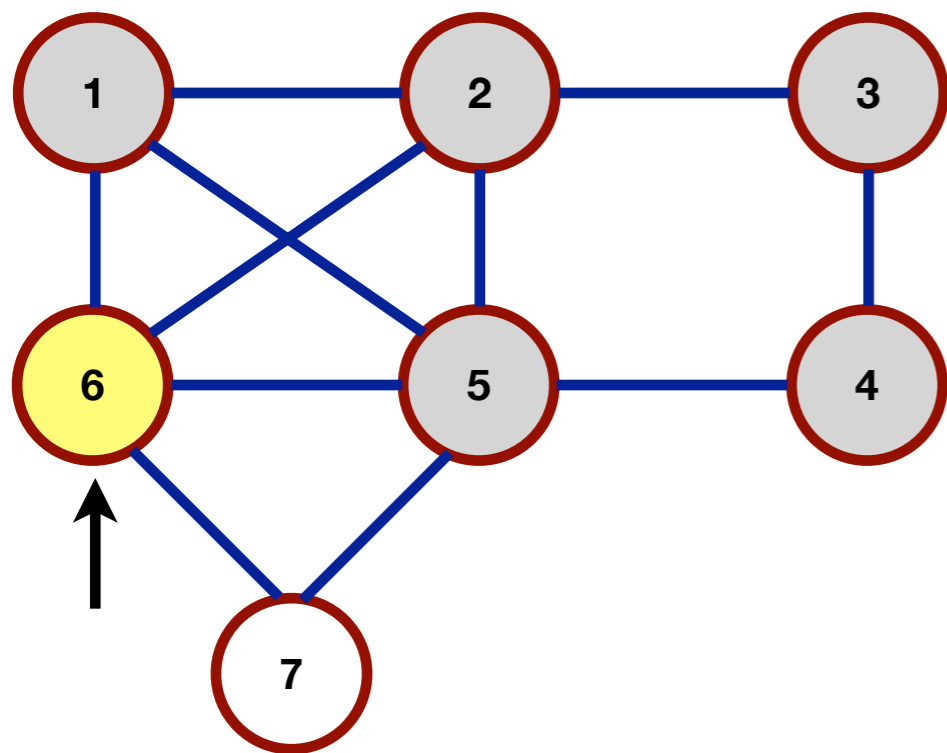
1 2 3 4 5



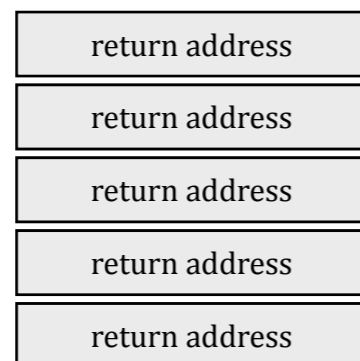
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



1 2 3 4 5

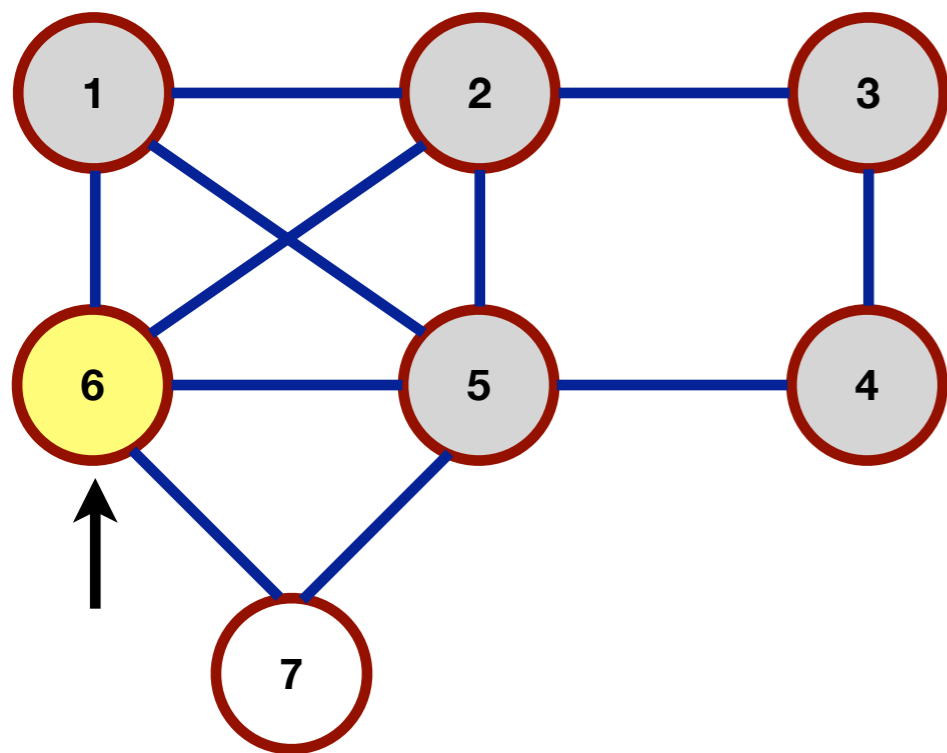


Runtime Stack

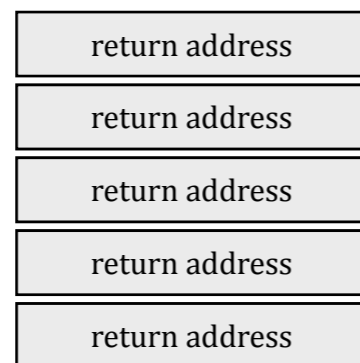
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

myGraph.DFS(vertex1)



1 2 3 4 5

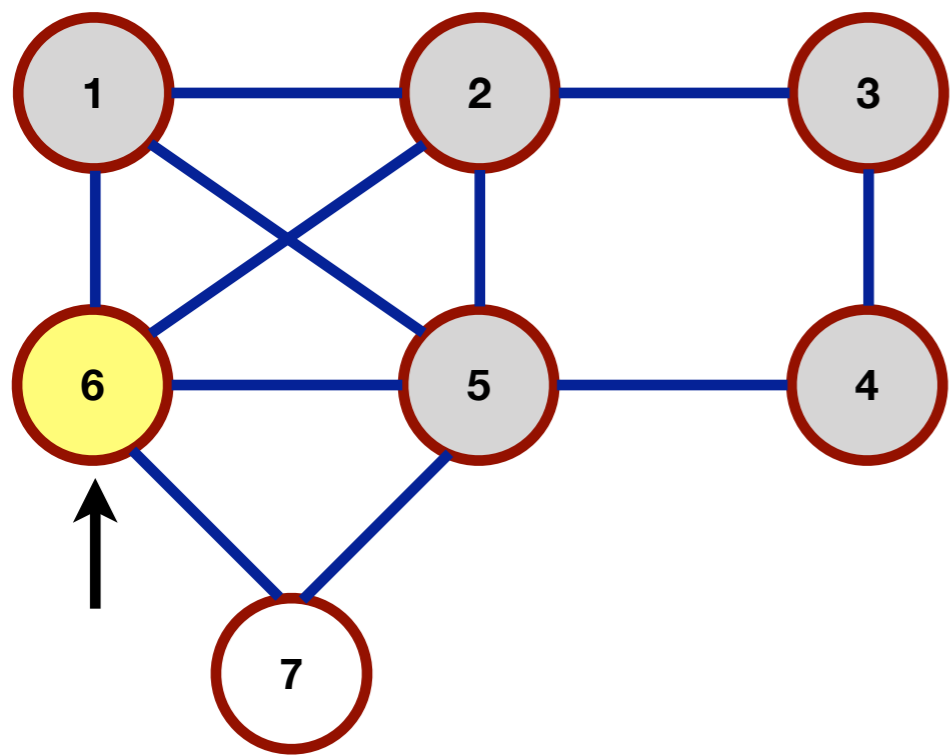


Runtime Stack

```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

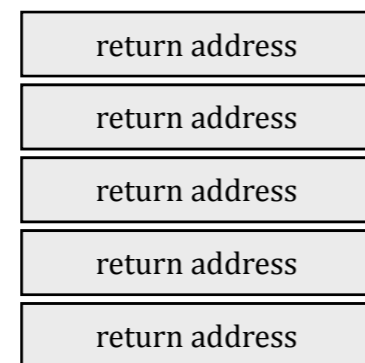

Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

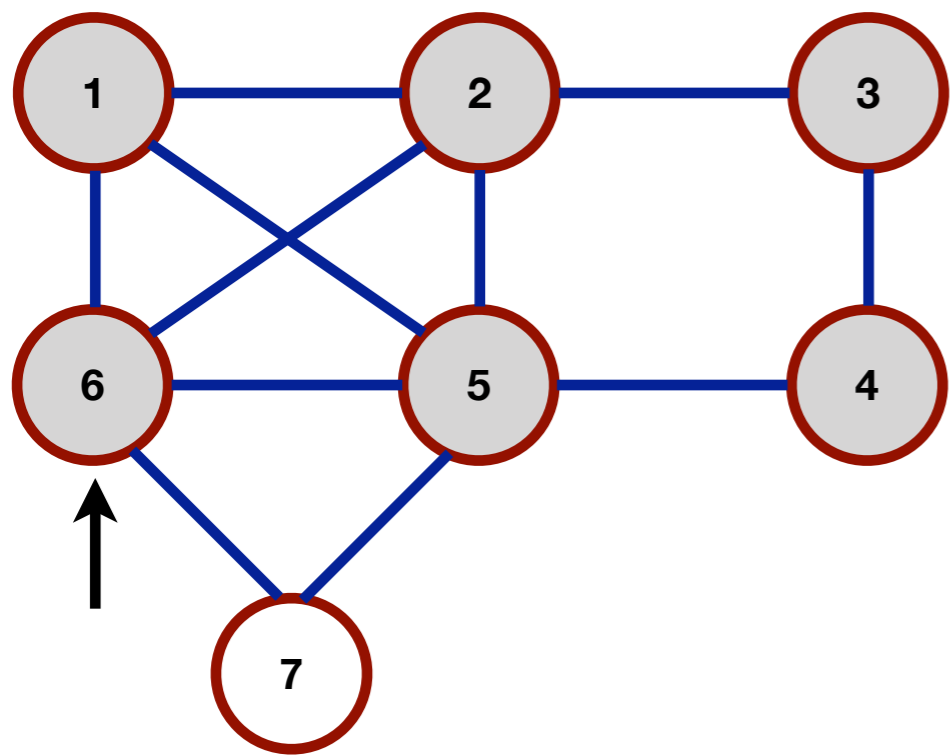
1 2 3 4 5 6



Runtime Stack

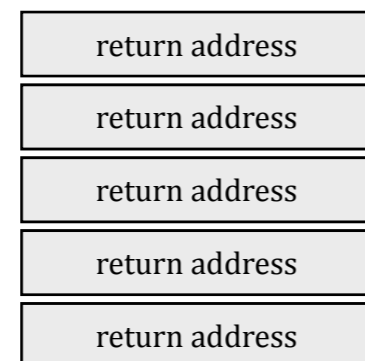
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

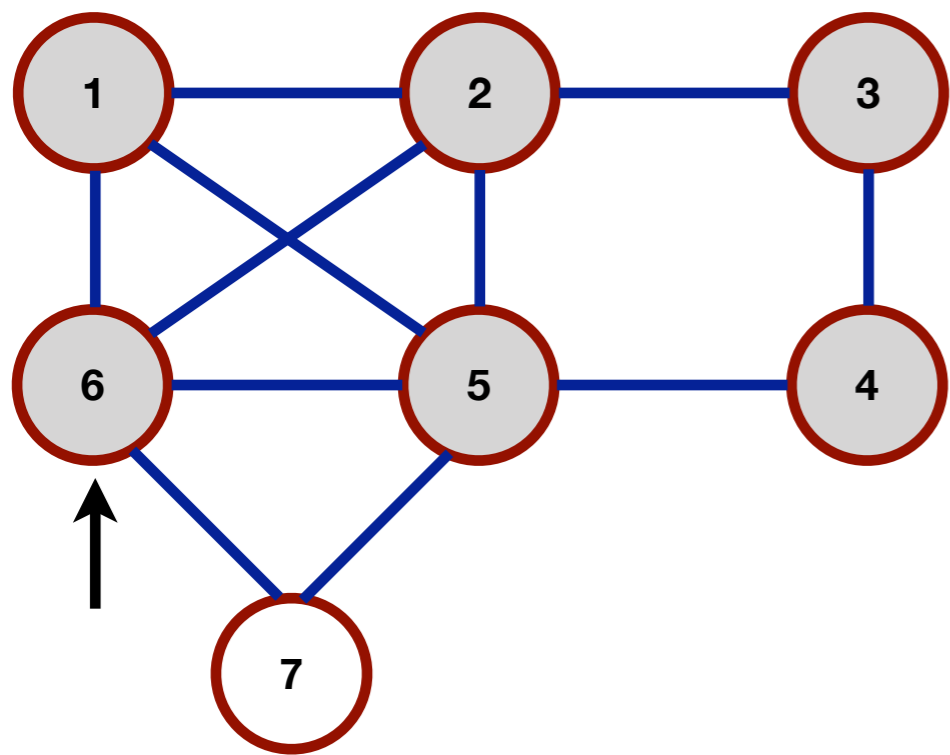
1 2 3 4 5 6



Runtime Stack

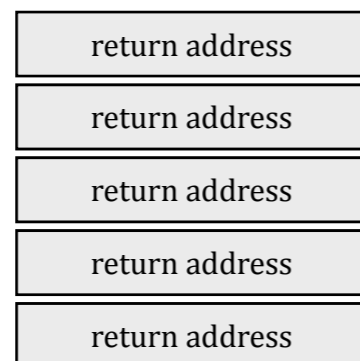
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

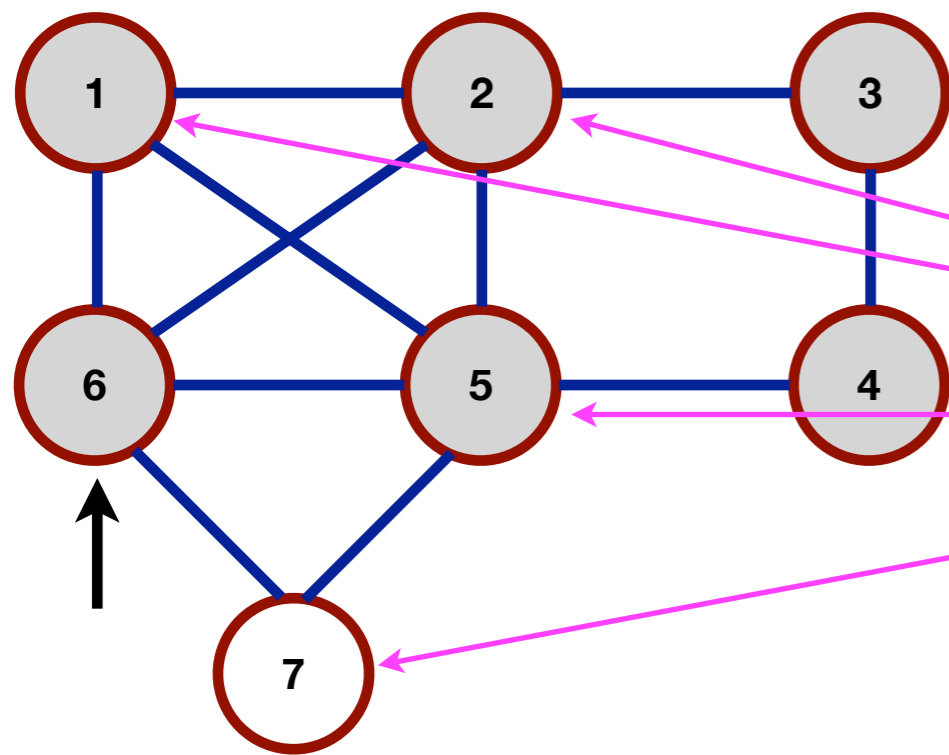
1 2 3 4 5 6



Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)

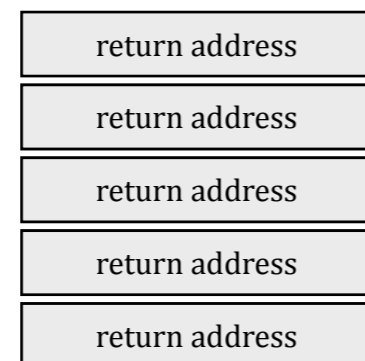


```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

```
[1] 2 5 6
[2] 1 3 5 6
[3] 2 4
[4] 3 5
[5] 1 2 4 6 7
[6] 1 2 5 7
[7] 5 6
```

Adjacency List

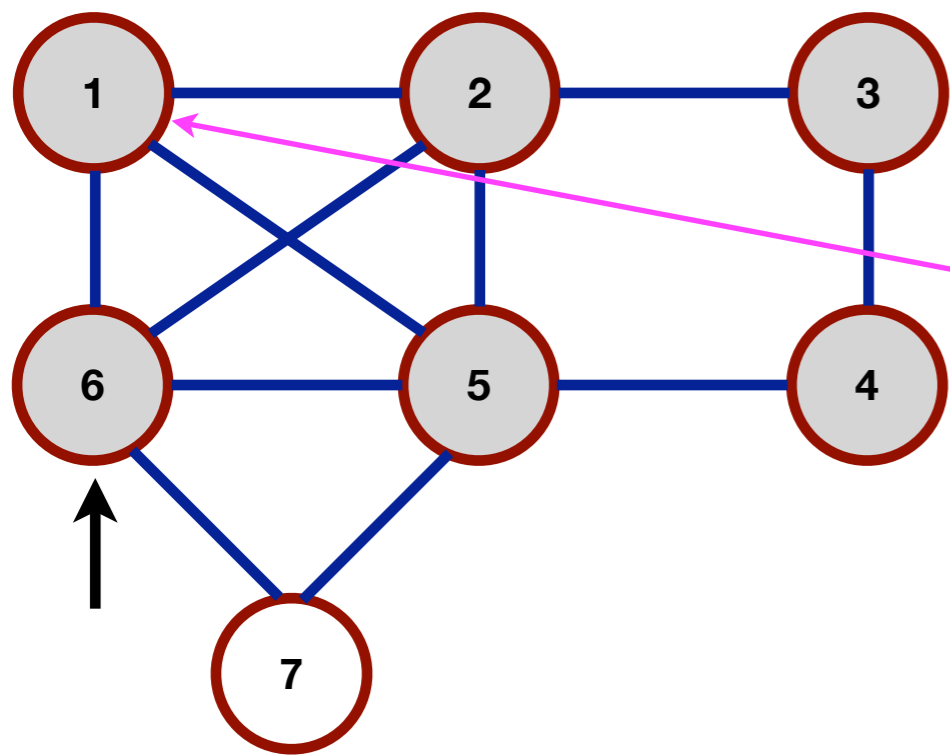
1 2 3 4 5 6



Runtime Stack

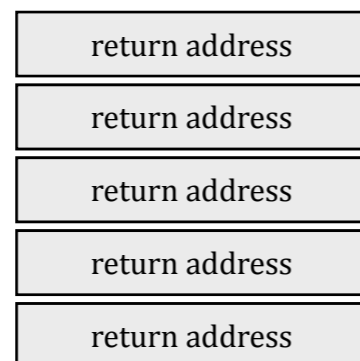
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

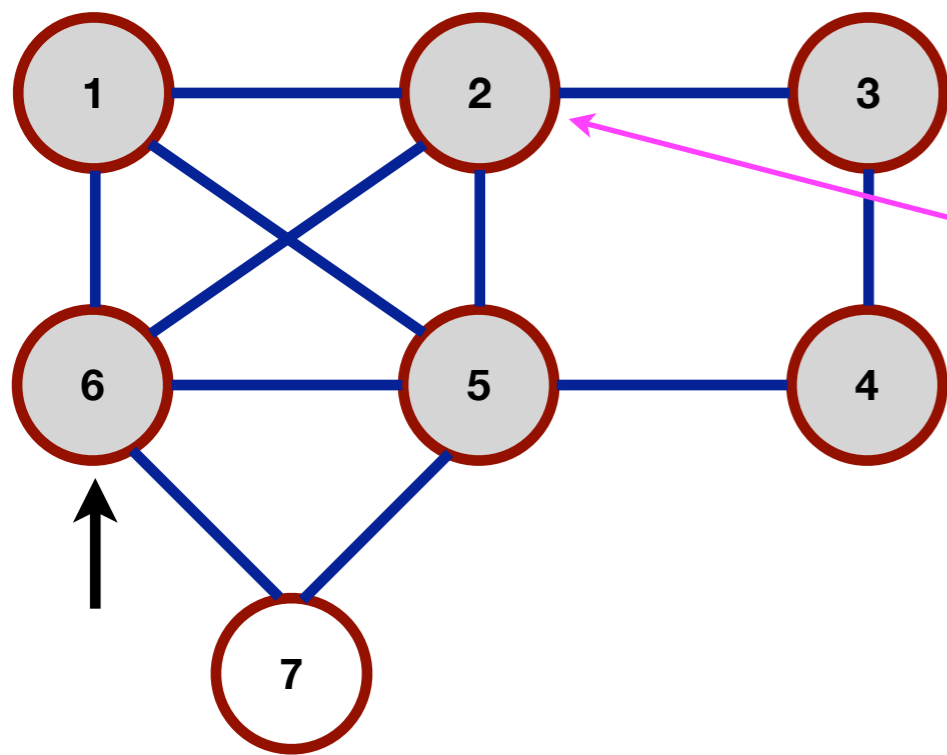
1 2 3 4 5 6



Runtime Stack

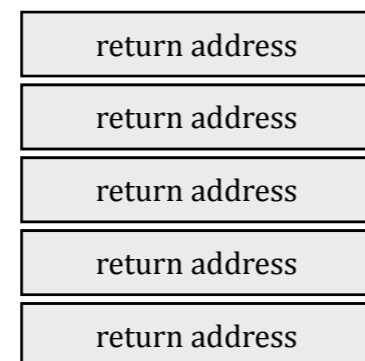
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

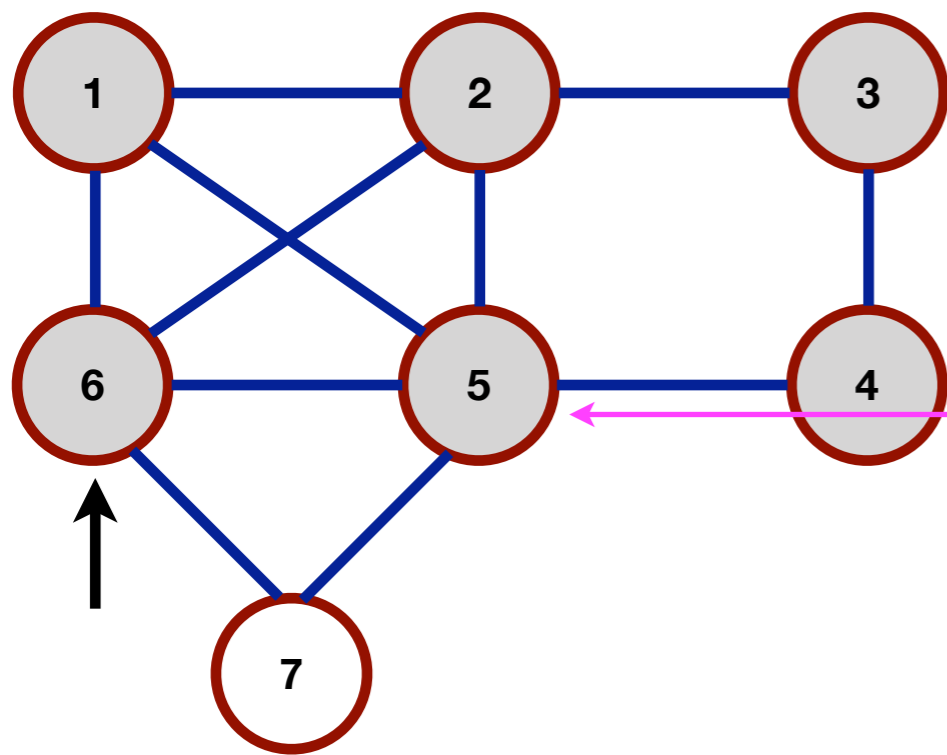
1 2 3 4 5 6



Runtime Stack

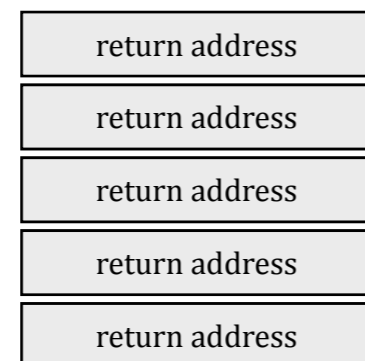
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

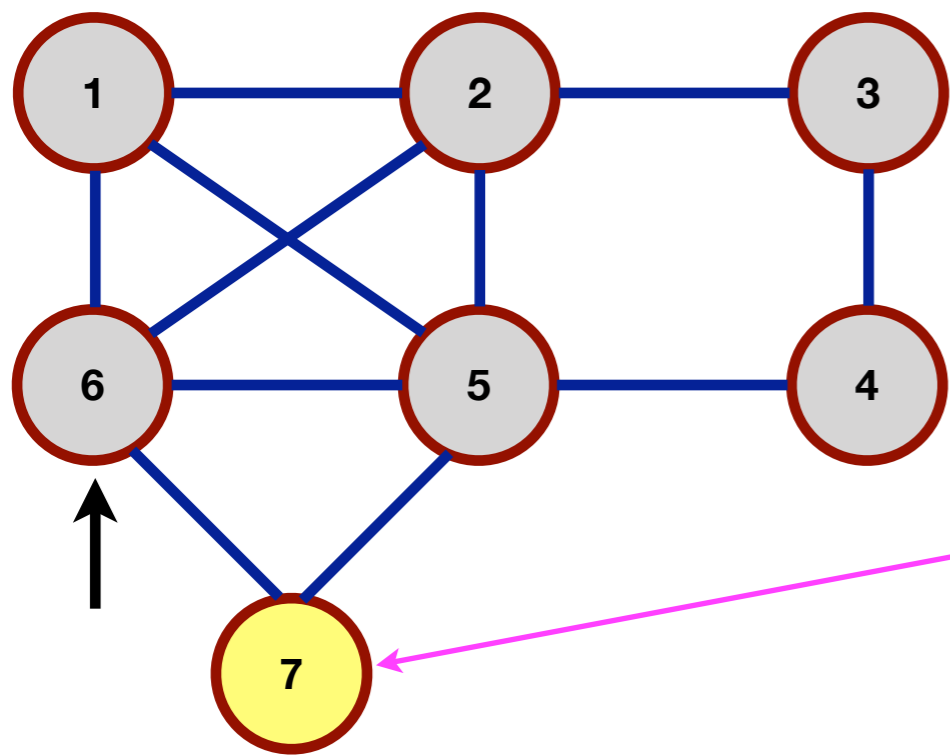
1 2 3 4 5 6



Runtime Stack

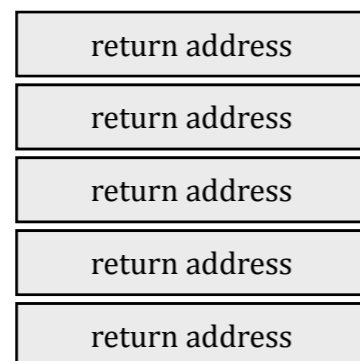
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

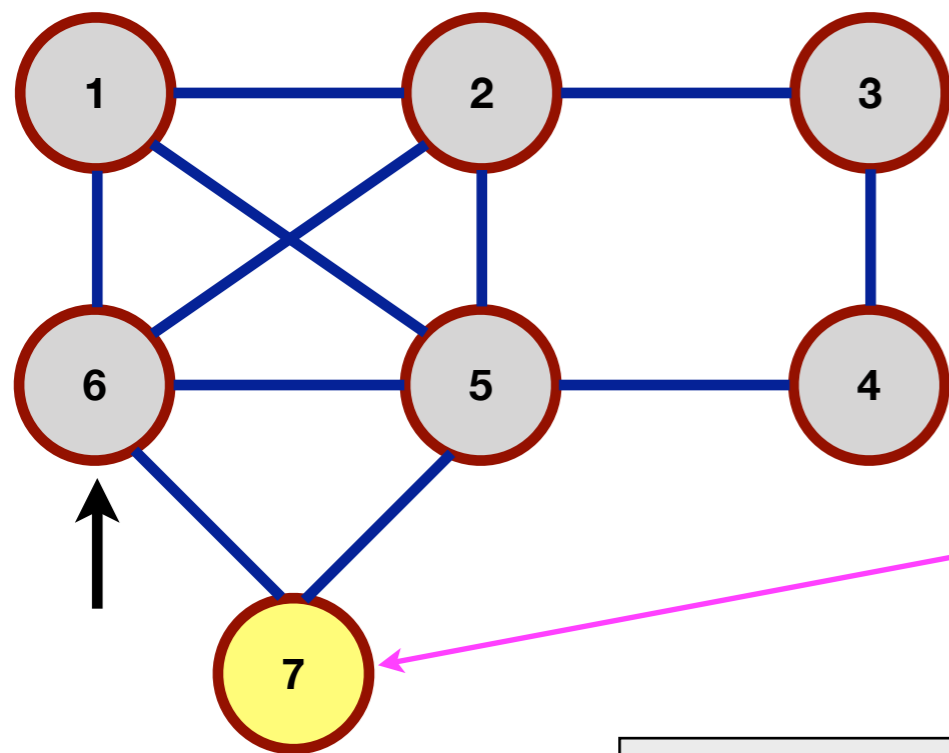
1 2 3 4 5 6



Runtime Stack

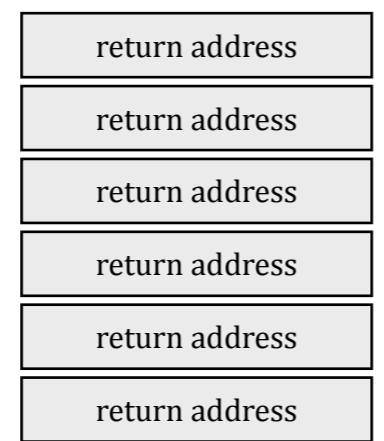
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

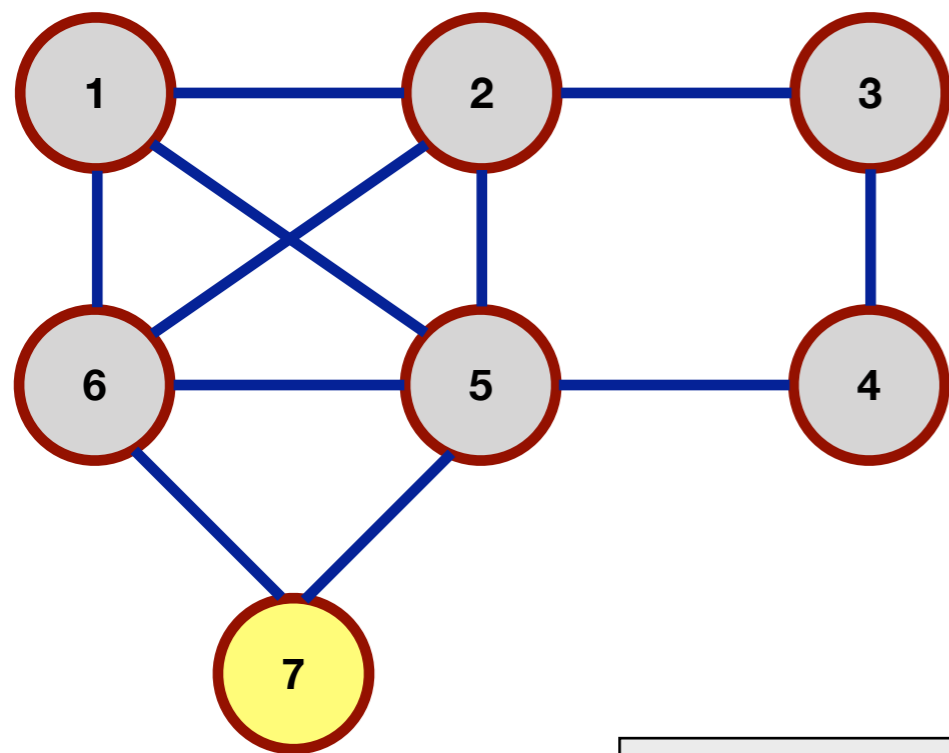
1 2 3 4 5 6



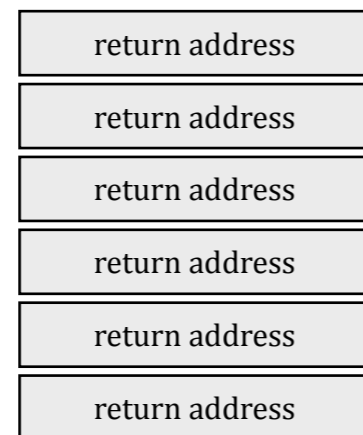
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



1 2 3 4 5 6

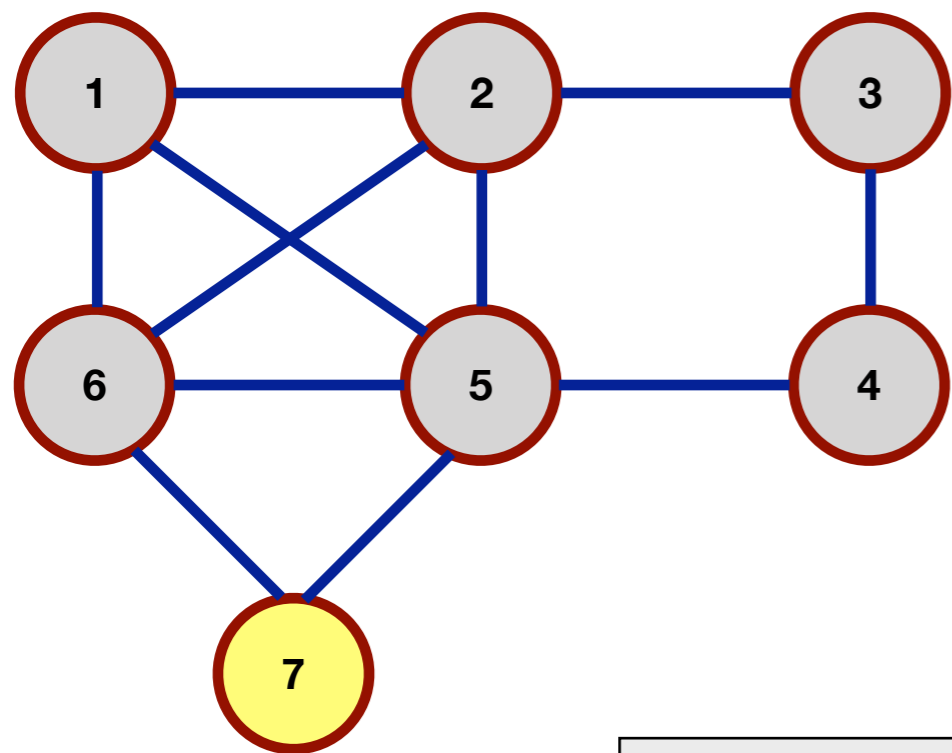


Runtime Stack

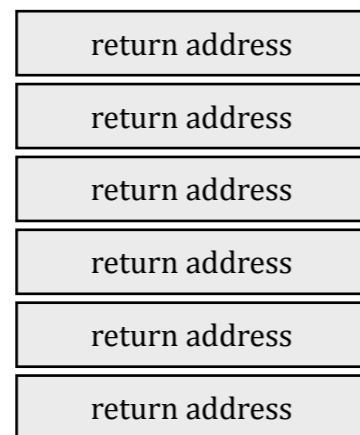
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

myGraph.DFS(vertex1)



1 2 3 4 5 6

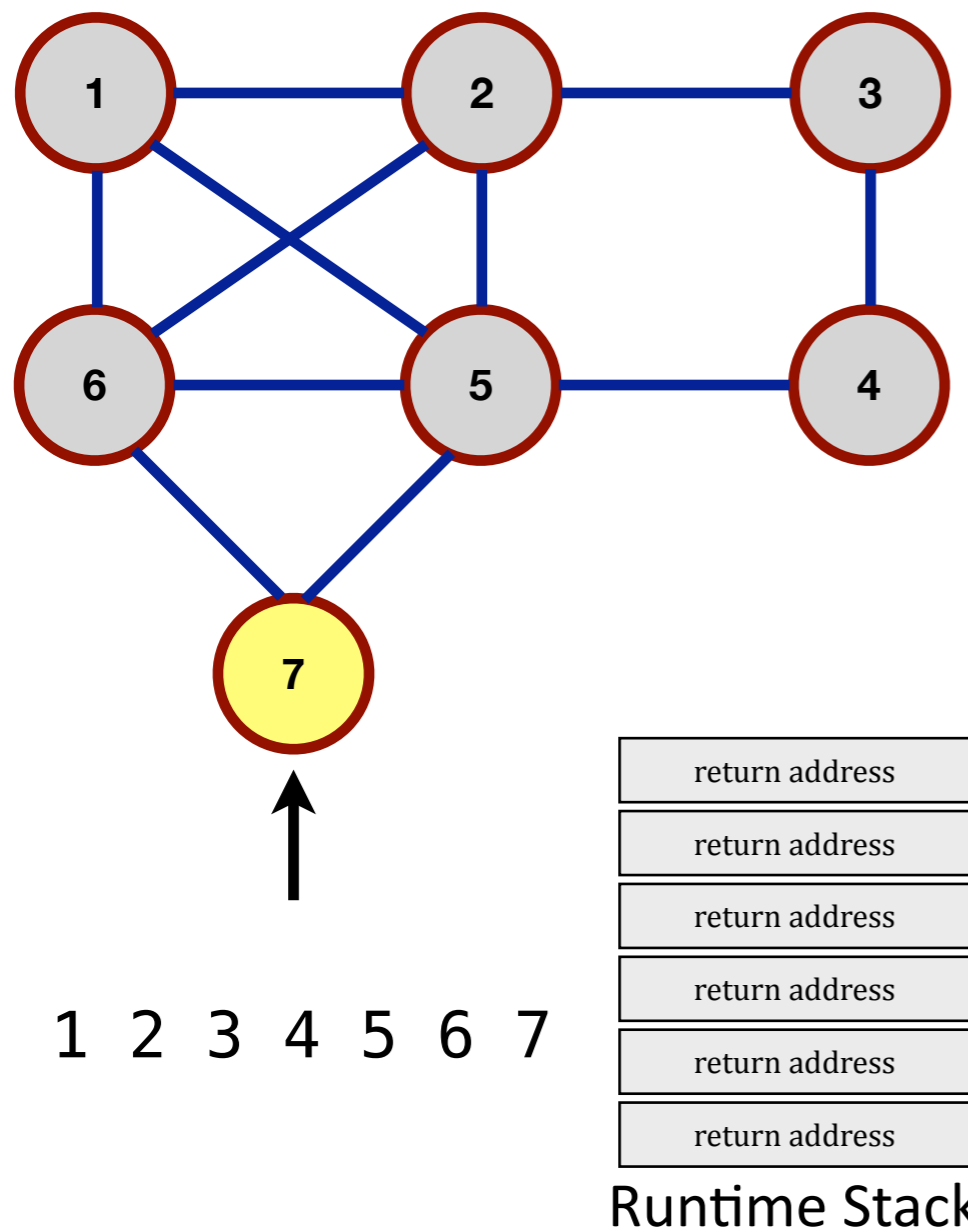


Runtime Stack

```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

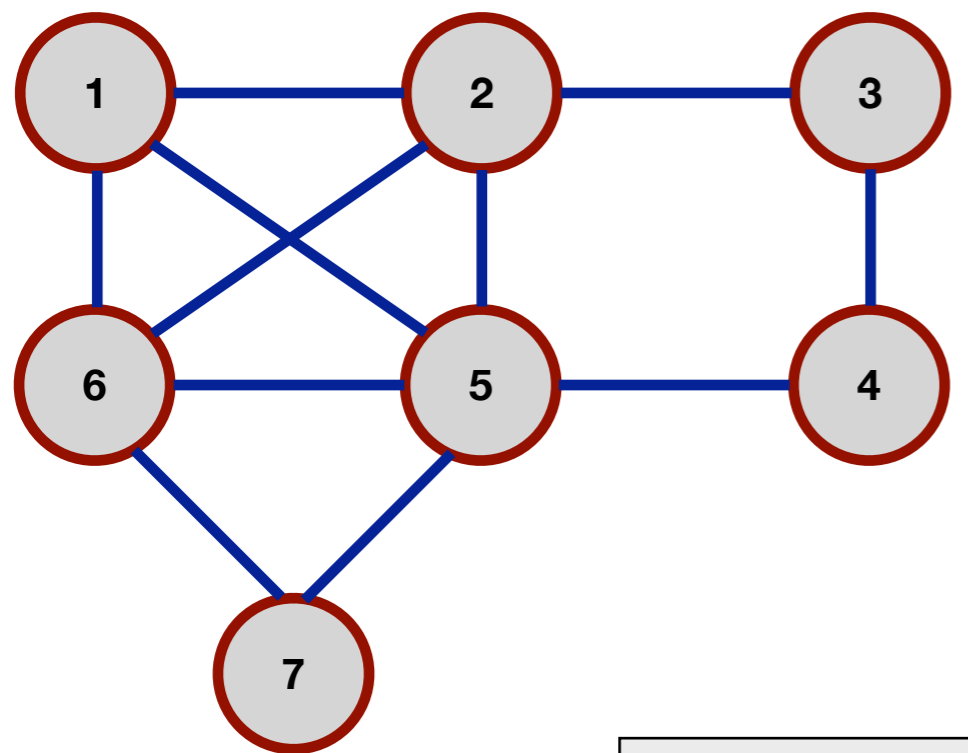
myGraph.DFS(vertex1)



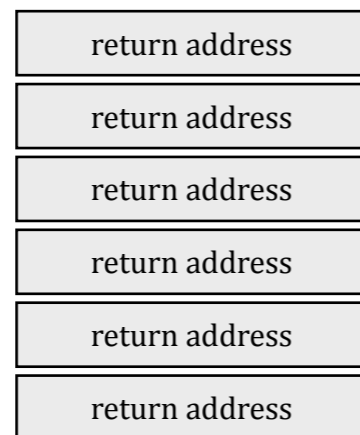
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

myGraph.DFS(vertex1)



1 2 3 4 5 6 7

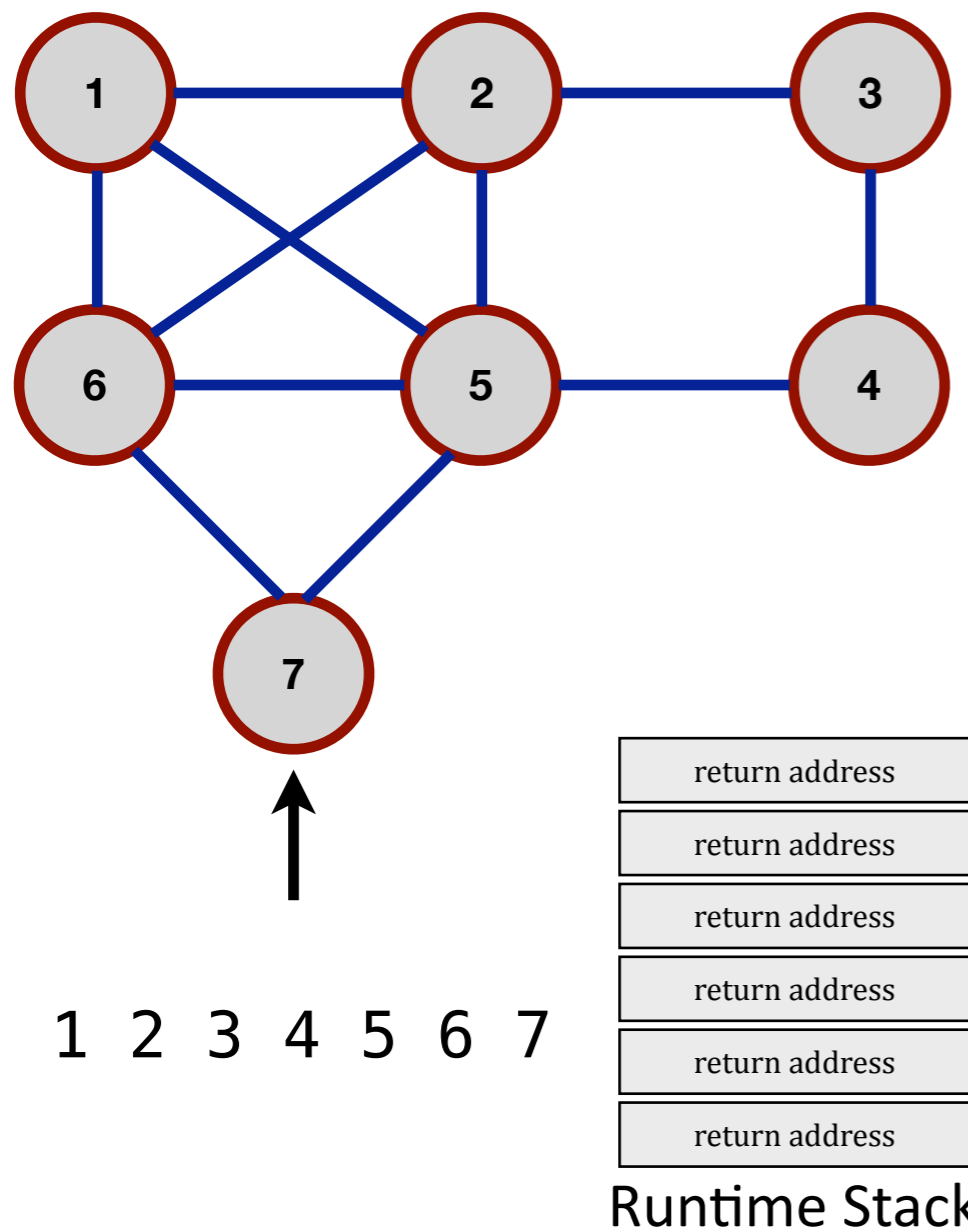


Runtime Stack

```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

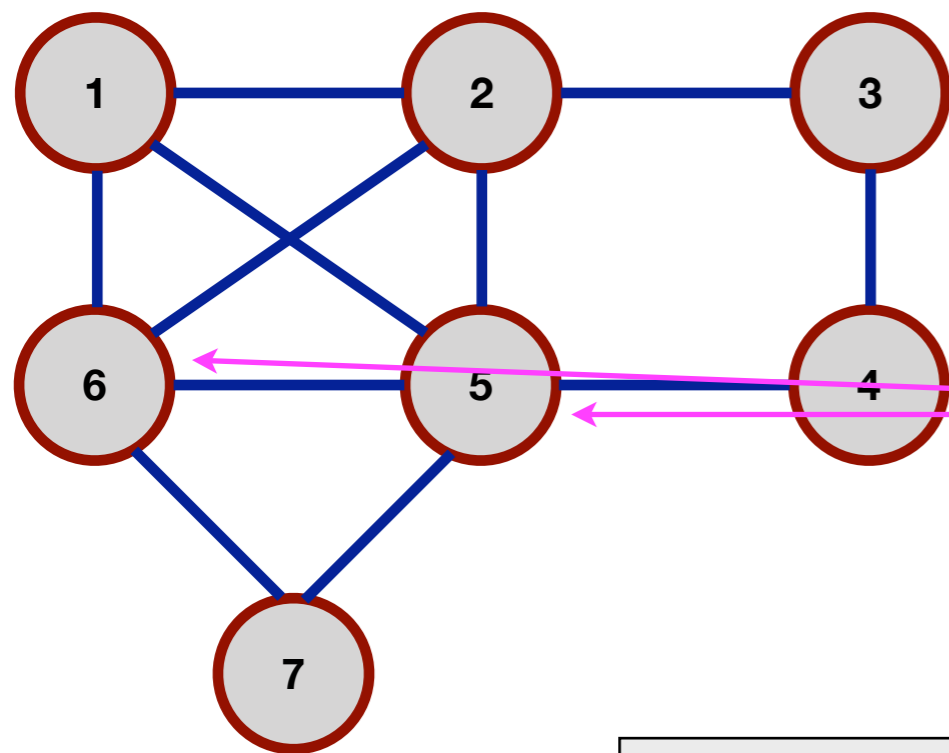
myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

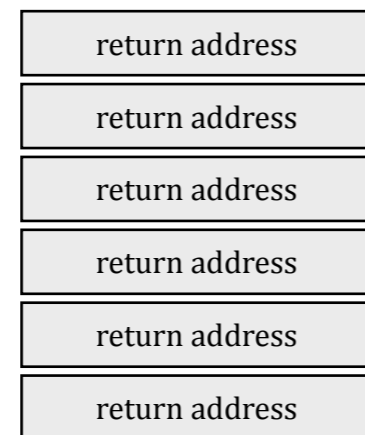
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7



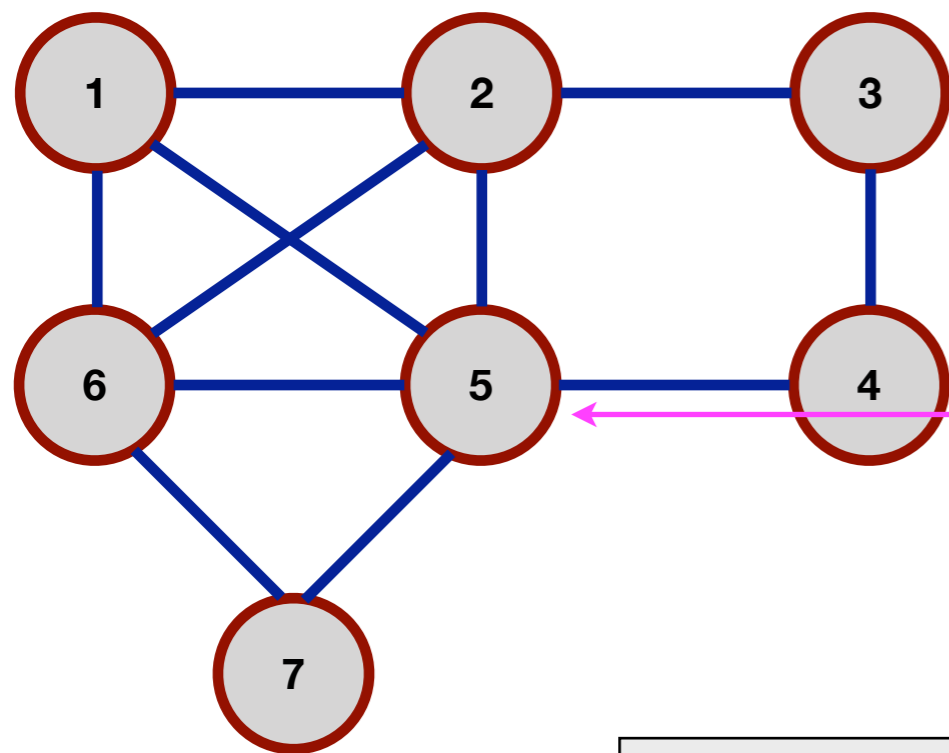
Runtime Stack

```
[1] 2 5 6
[2] 1 3 5 6
[3] 2 4
[4] 3 5
[5] 1 2 4 6 7
[6] 1 2 5 7
[7] 5 6
```

Adjacency List

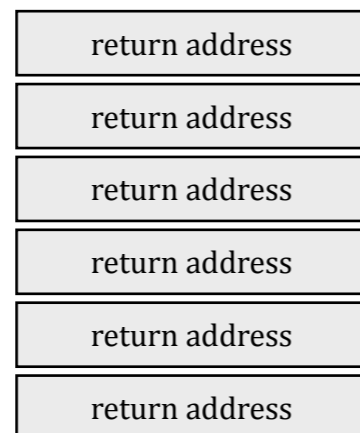
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

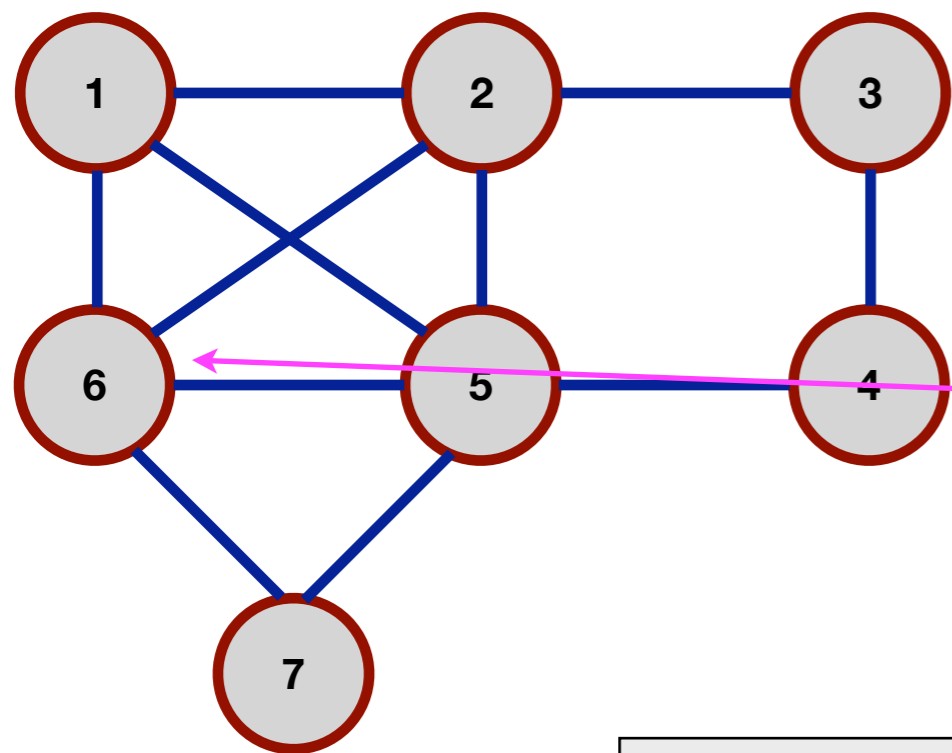
1 2 3 4 5 6 7



Runtime Stack

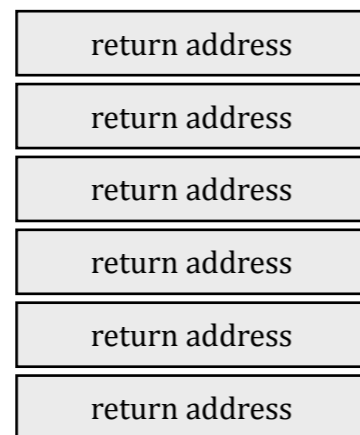
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

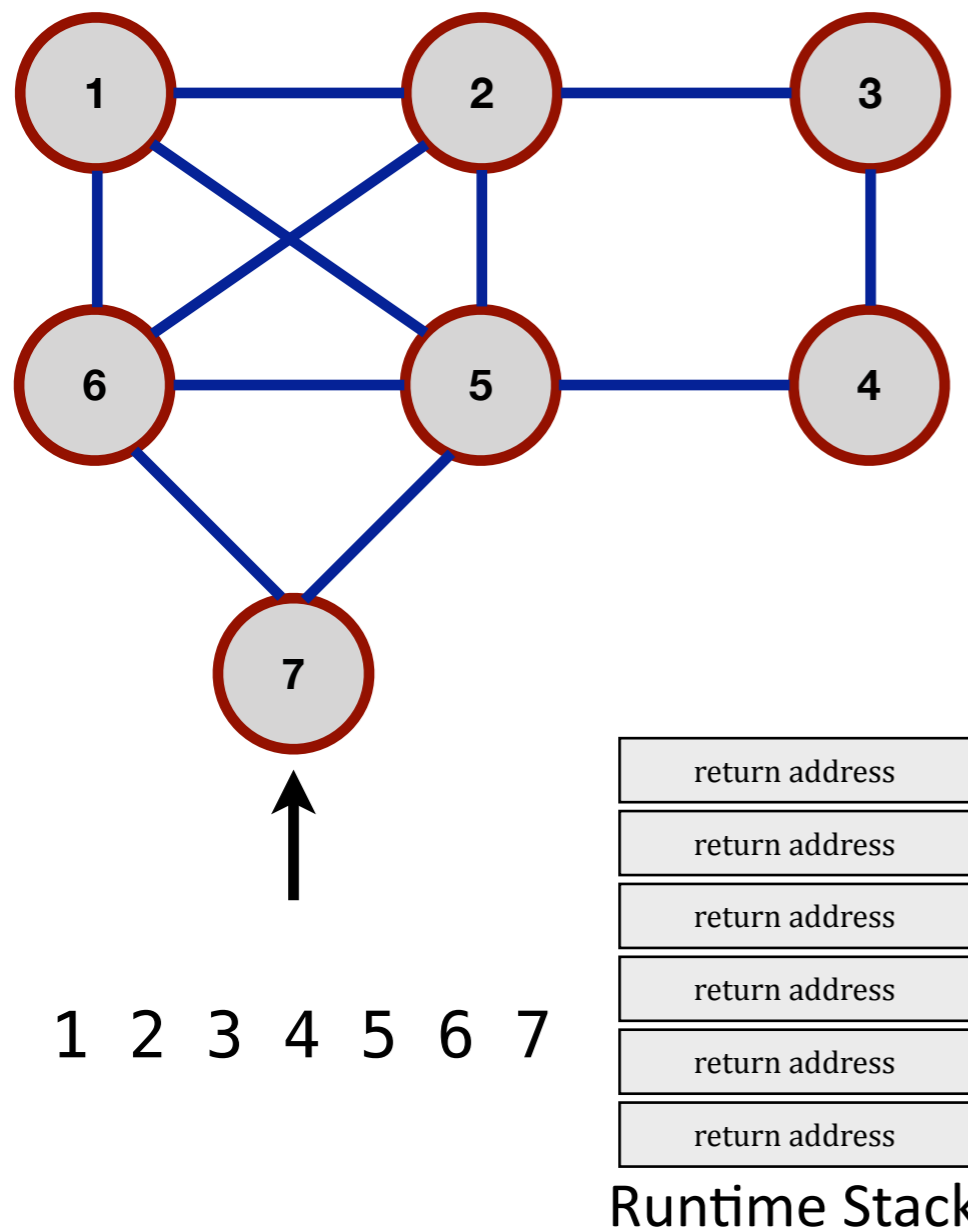
1 2 3 4 5 6 7



Runtime Stack

Depth-First Search / Traversal

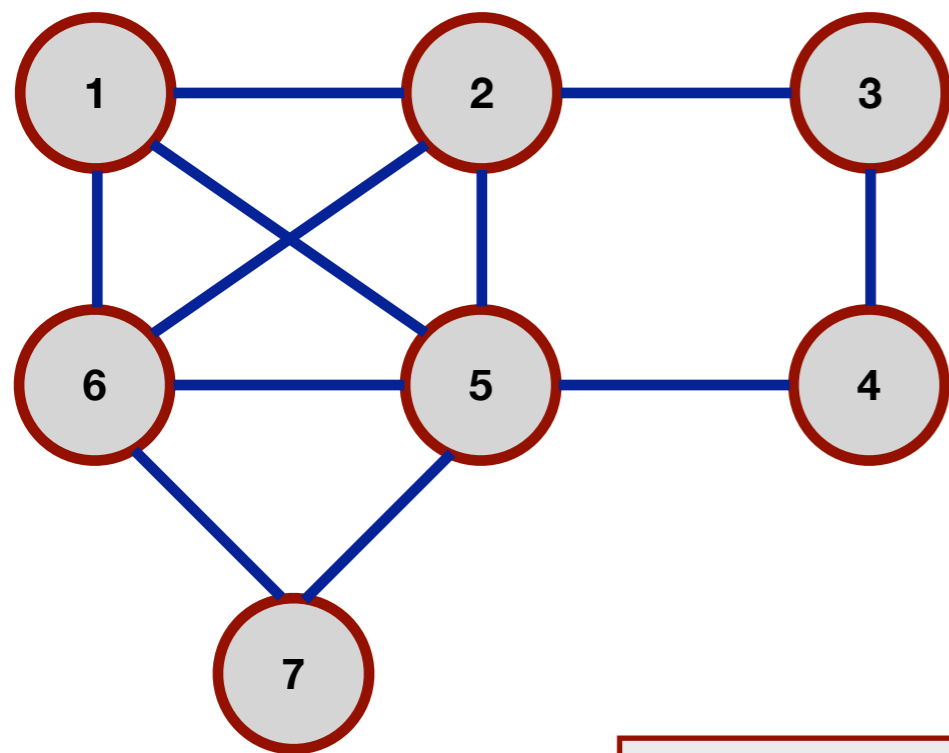
myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

Depth-First Search / Traversal

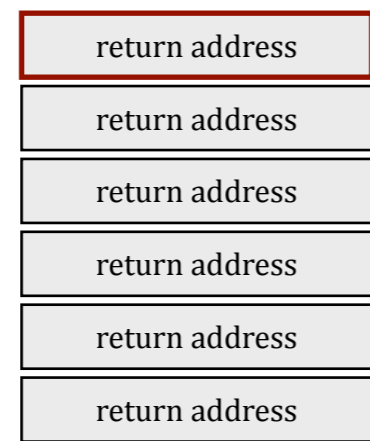
myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

endproc

1 2 3 4 5 6 7

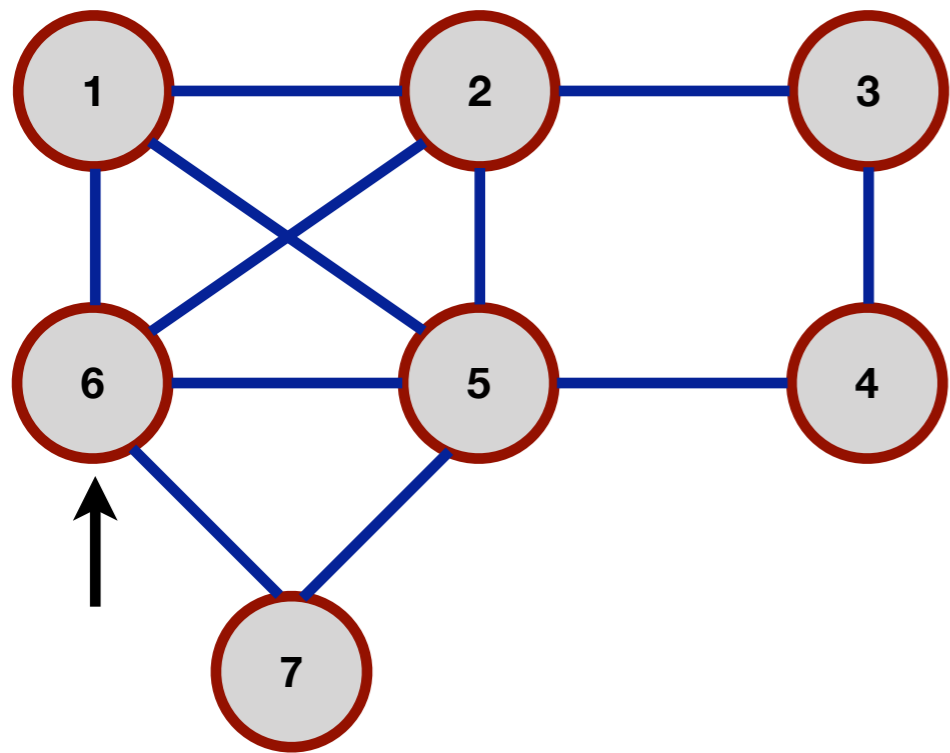


Runtime Stack

End of most recent recursive call.
Pop return address off the runtime stack and return.

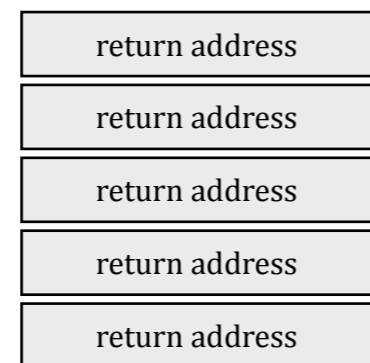
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

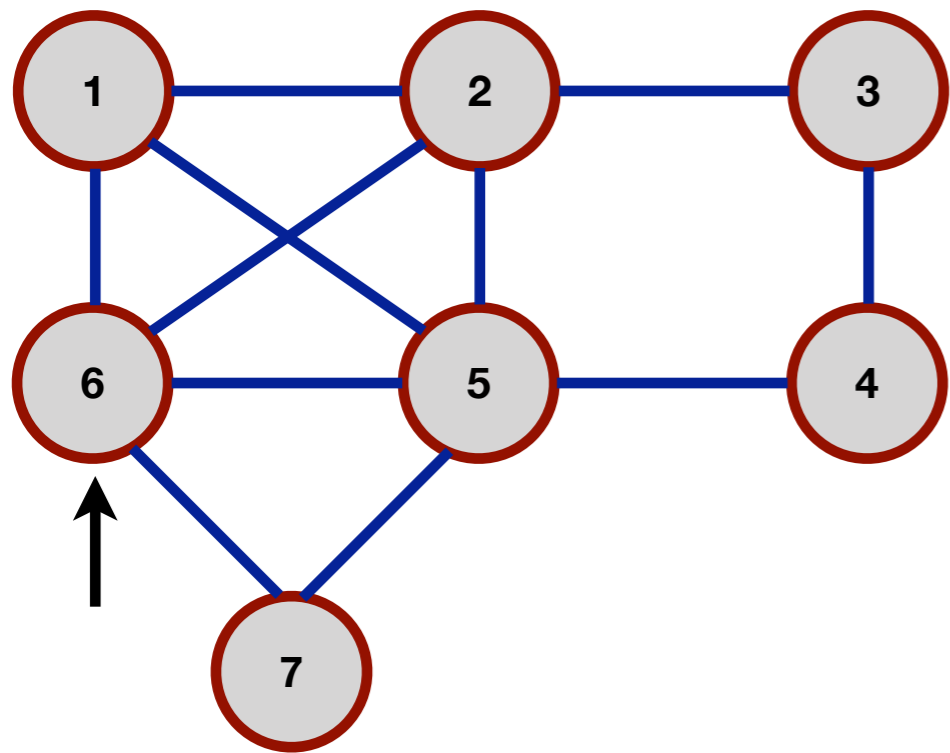
1 2 3 4 5 6 7



Runtime Stack

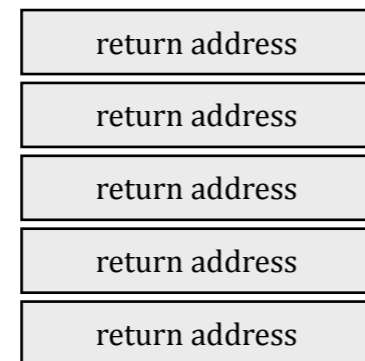
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

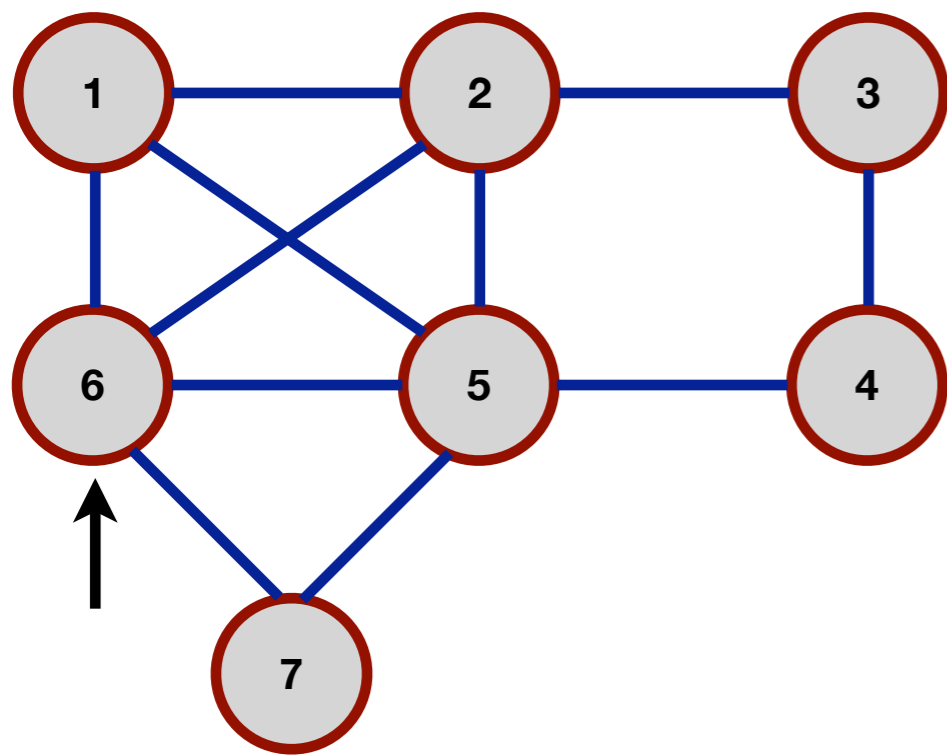
1 2 3 4 5 6 7



Runtime Stack

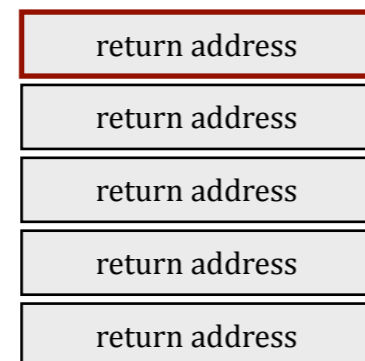
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

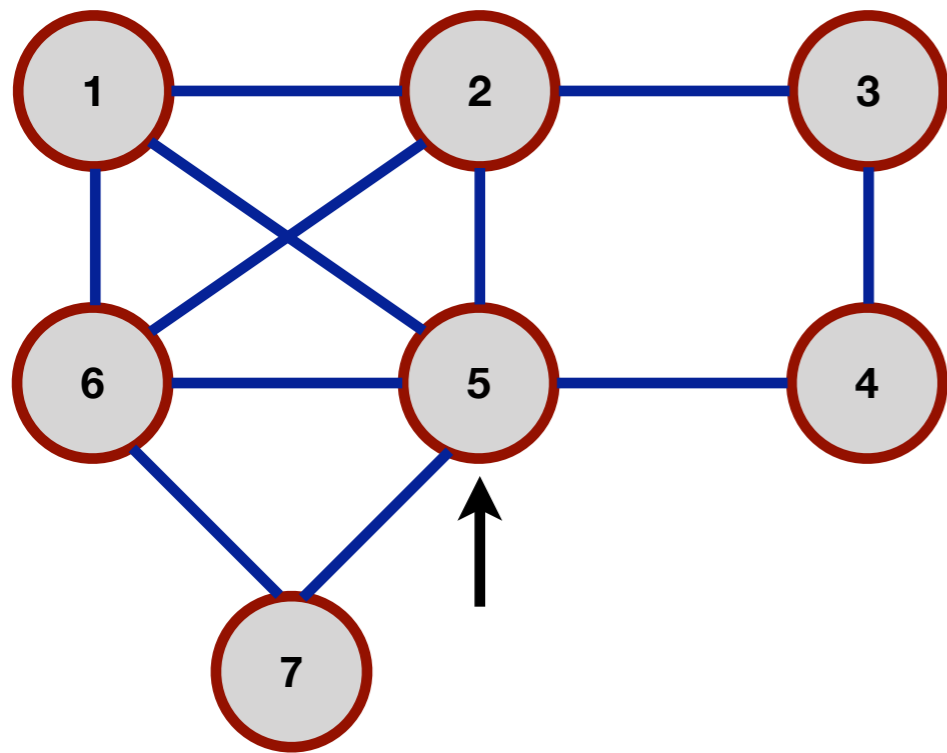


Runtime Stack

Pop return address off the runtime stack and return.

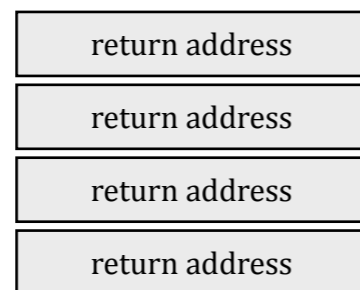
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

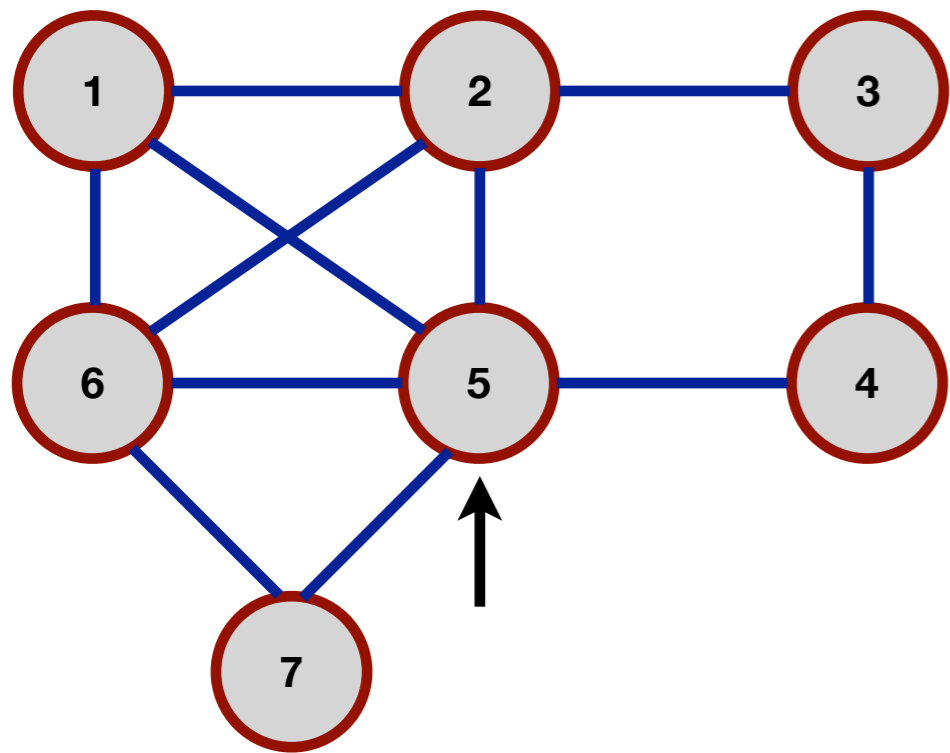
1 2 3 4 5 6 7



Runtime Stack

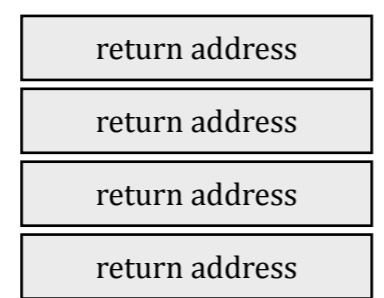
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

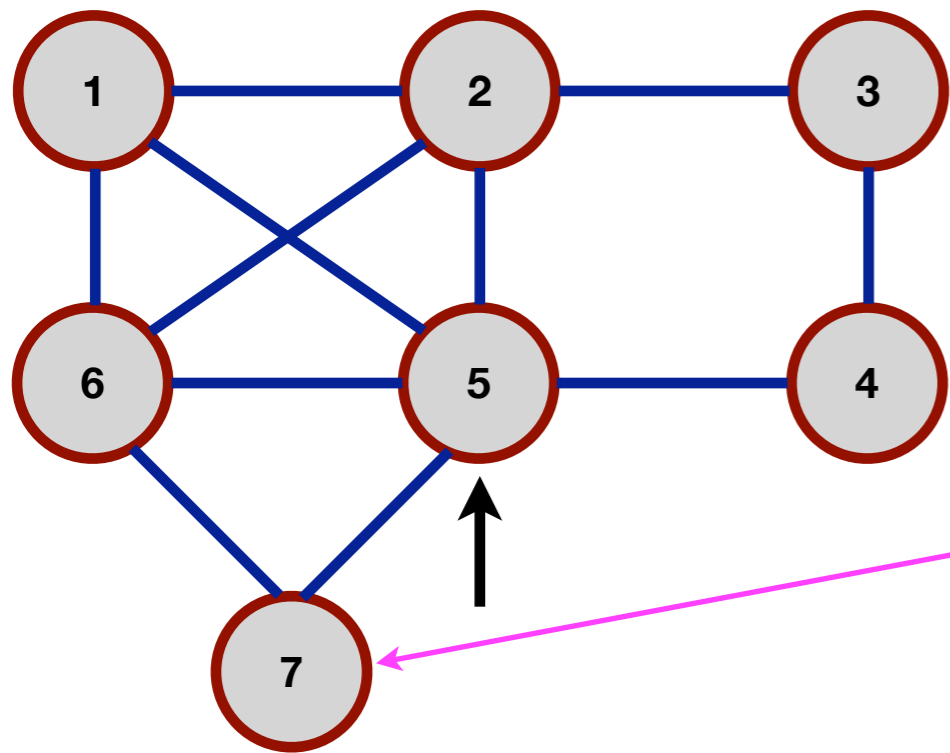
1 2 3 4 5 6 7



Runtime Stack

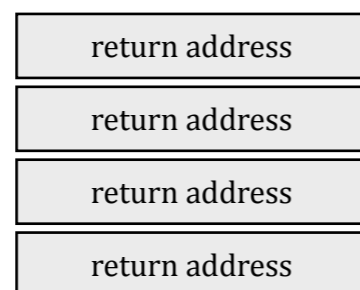
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

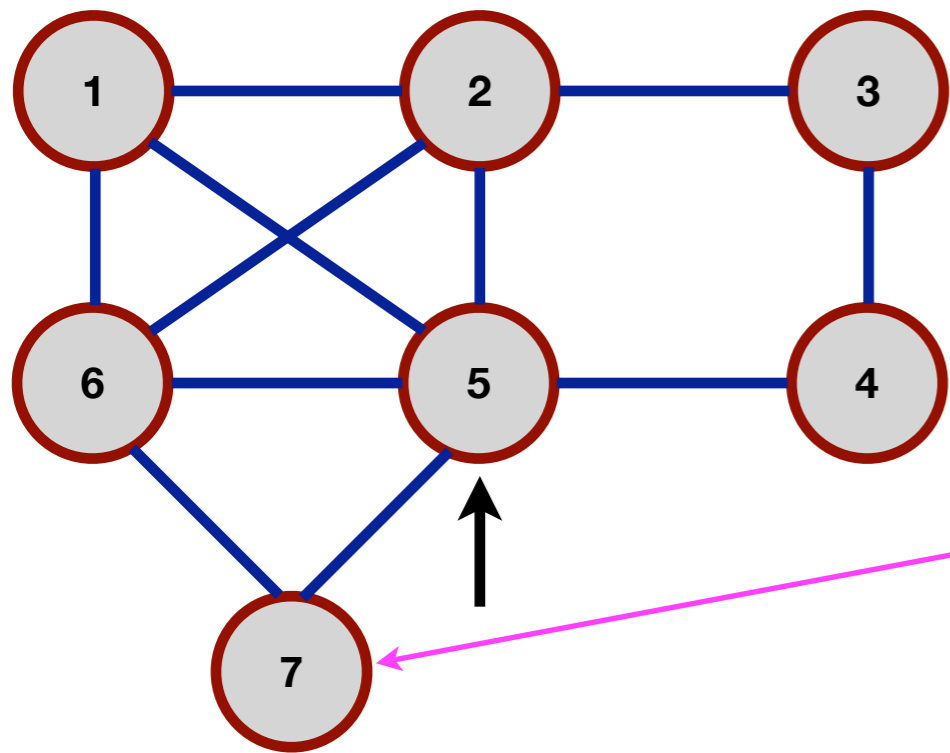
1 2 3 4 5 6 7



Runtime Stack

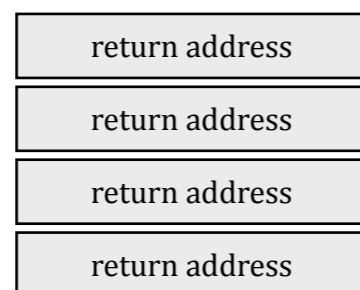
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

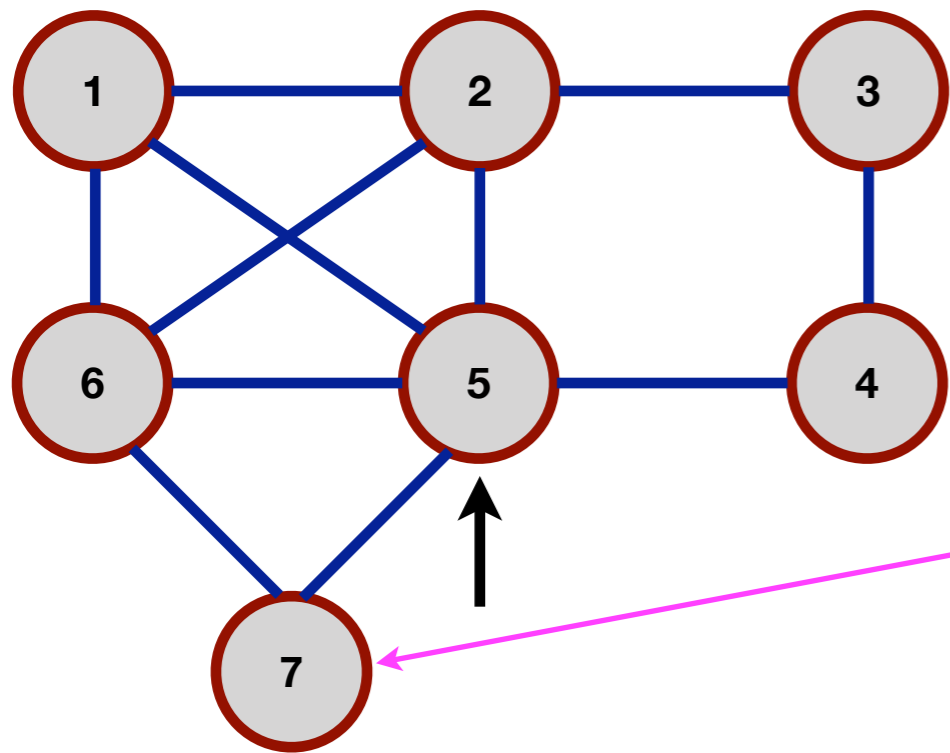
1 2 3 4 5 6 7



Runtime Stack

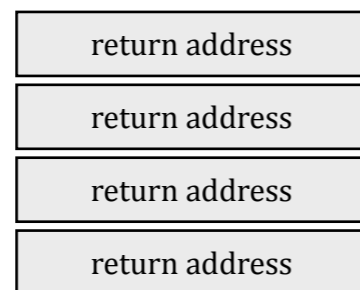
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

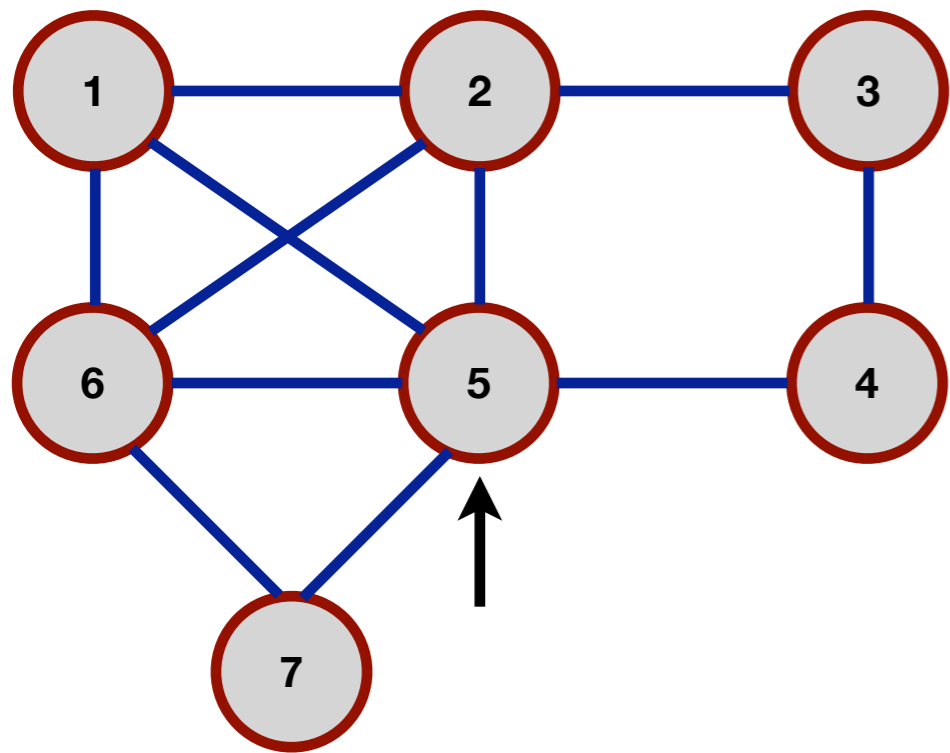
1 2 3 4 5 6 7



Runtime Stack

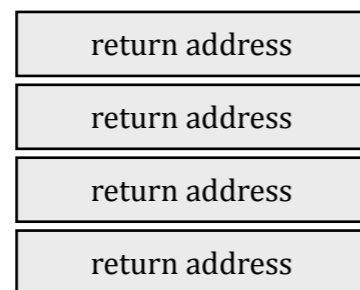
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

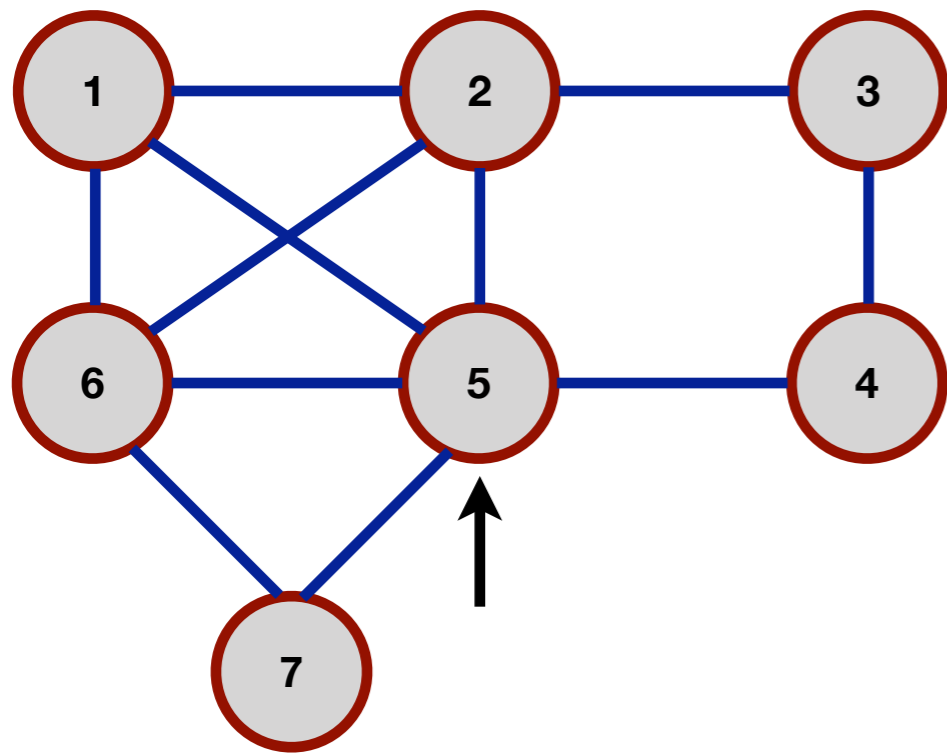
1 2 3 4 5 6 7



Runtime Stack

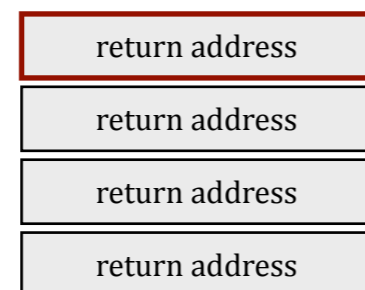
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

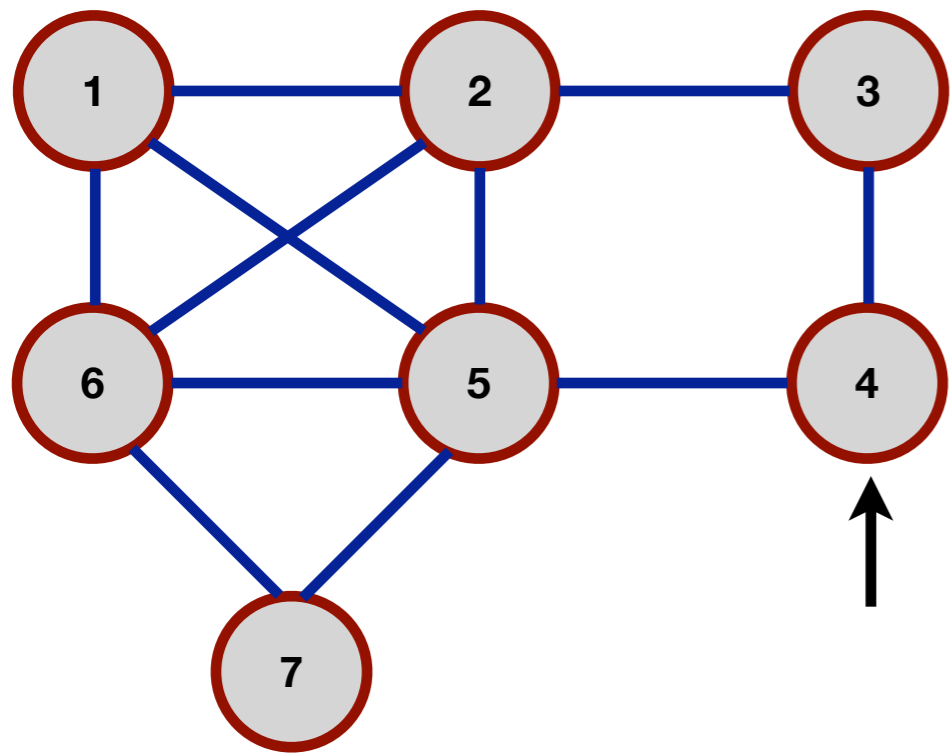
1 2 3 4 5 6 7



Runtime Stack

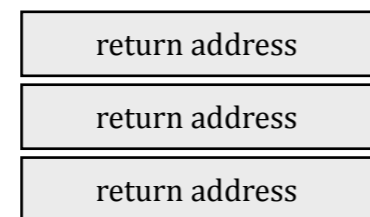
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

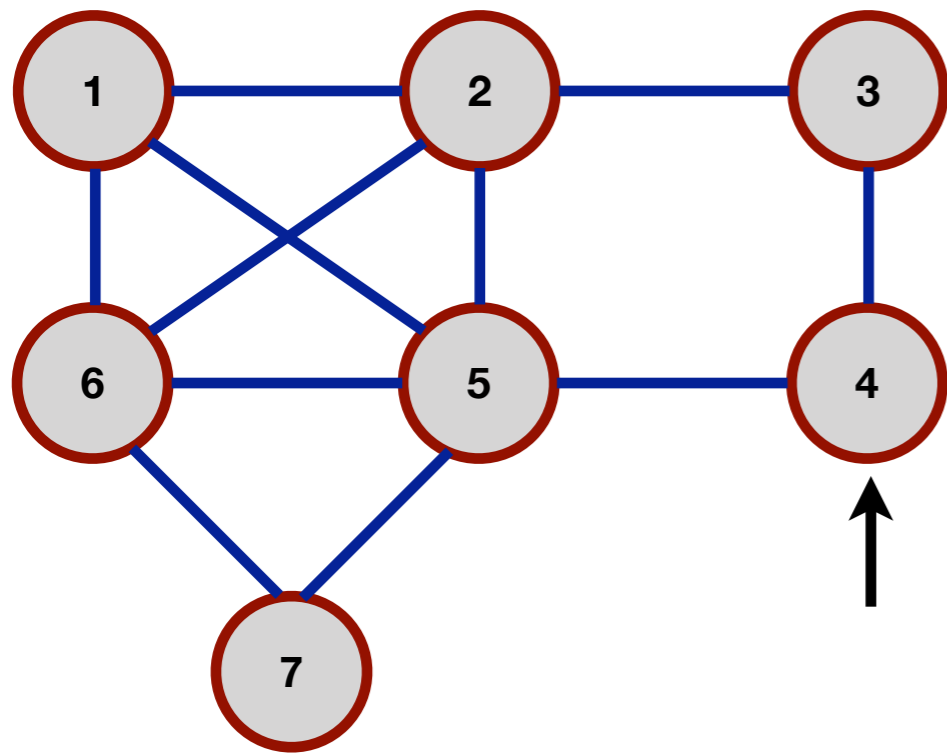
1 2 3 4 5 6 7



Runtime Stack

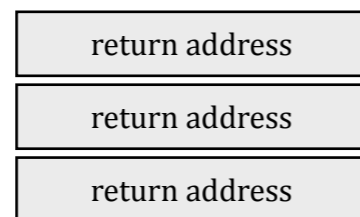
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

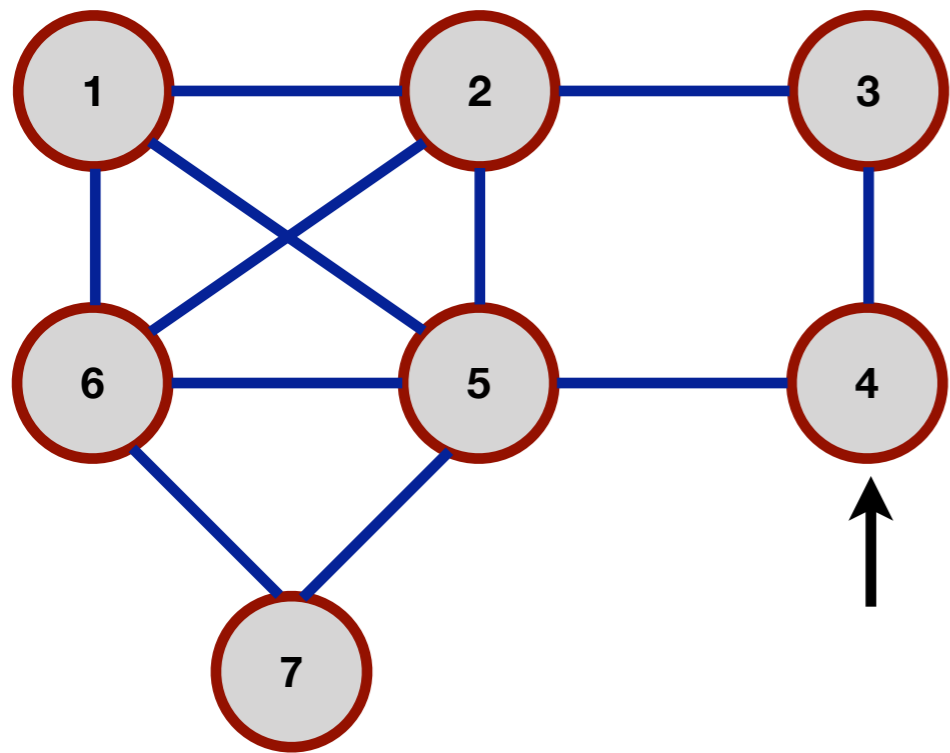
1 2 3 4 5 6 7



Runtime Stack

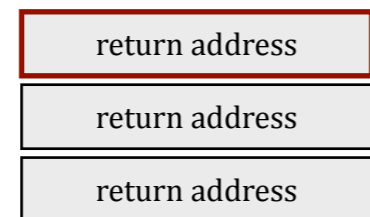
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

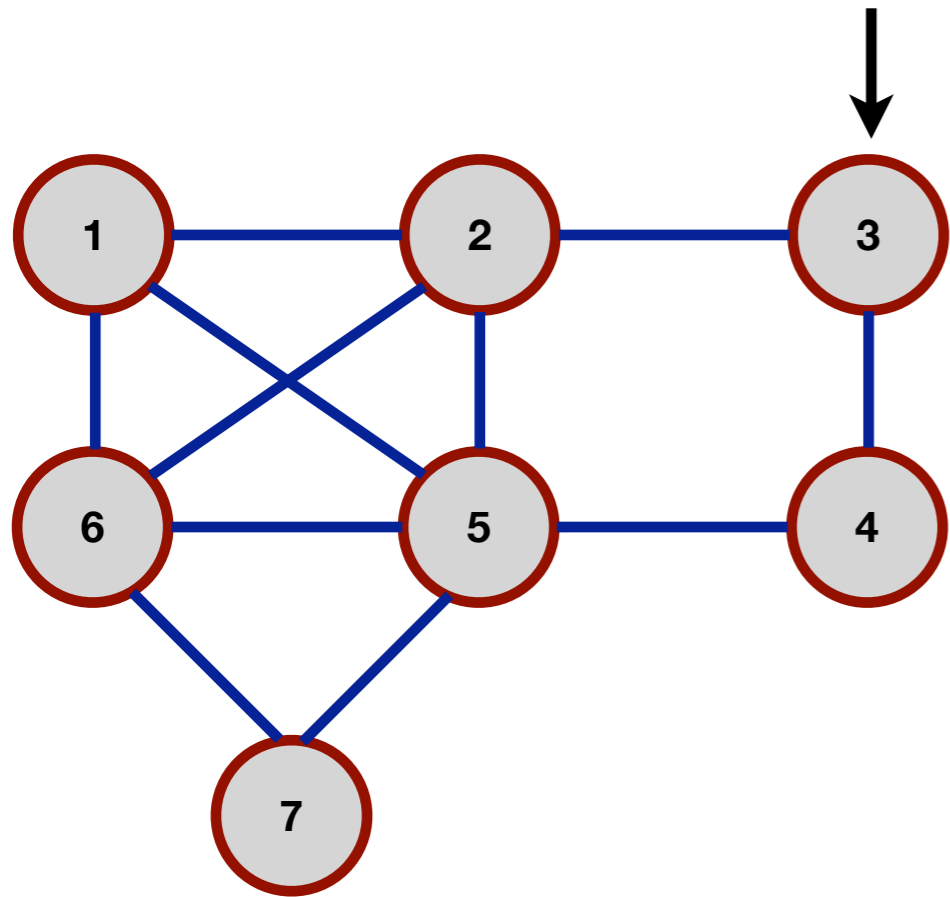
1 2 3 4 5 6 7



Runtime Stack

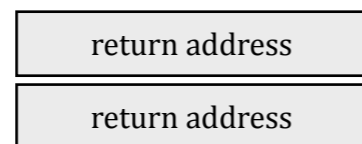
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

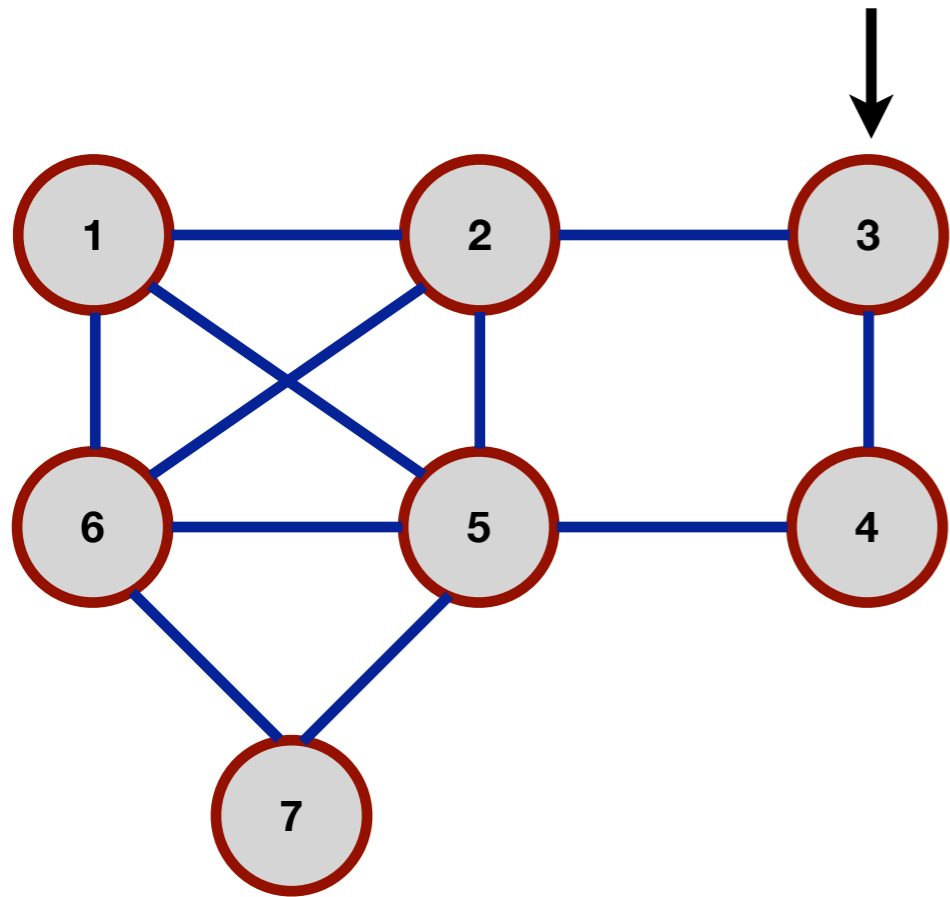
1 2 3 4 5 6 7



Runtime Stack

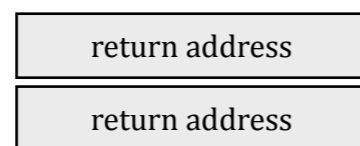
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

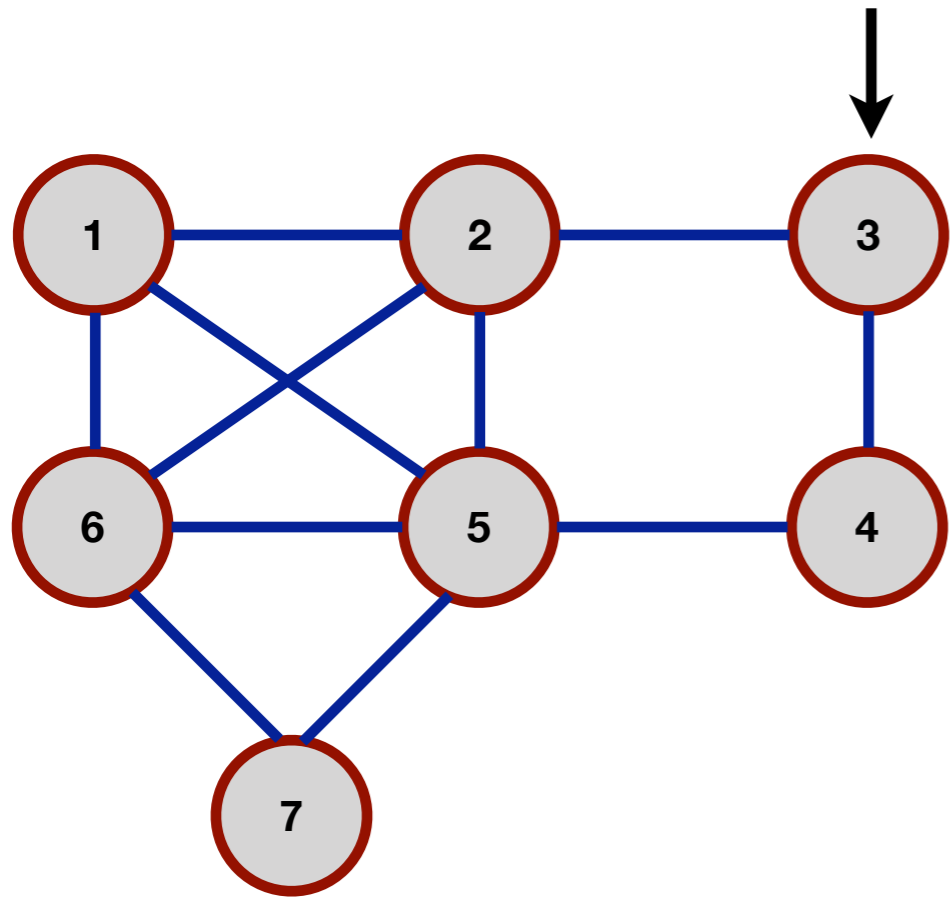
1 2 3 4 5 6 7



Runtime Stack

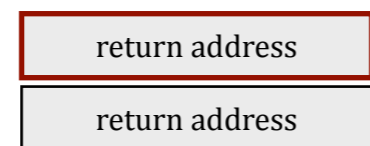
Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

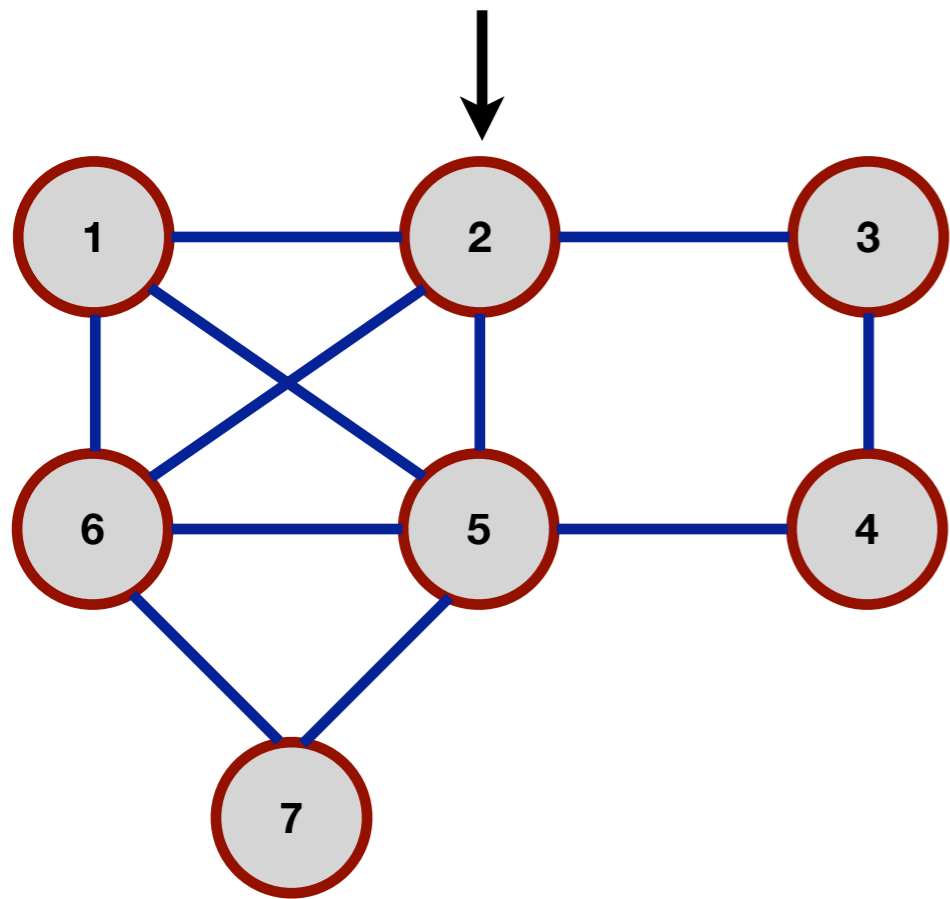
1 2 3 4 5 6 7



Runtime Stack

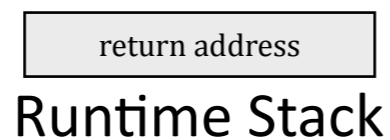
Depth-First Search / Traversal

myGraph.DFS(vertex1)



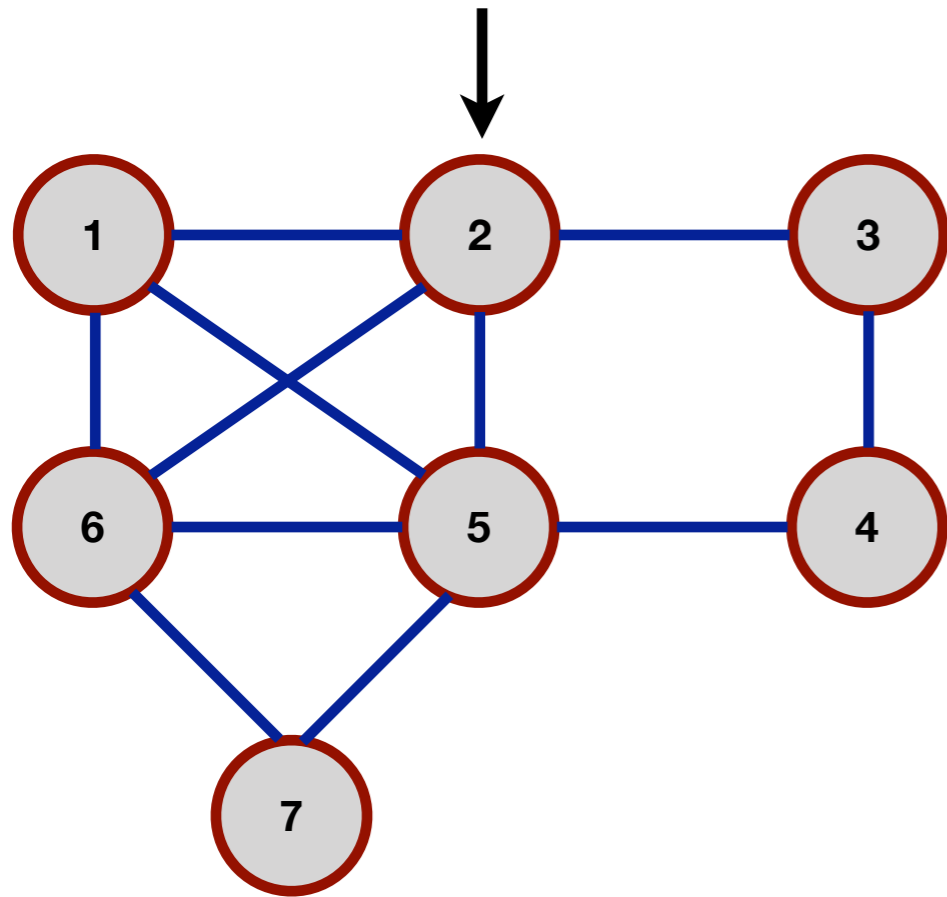
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7



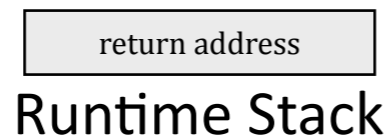
Depth-First Search / Traversal

myGraph.DFS(vertex1)



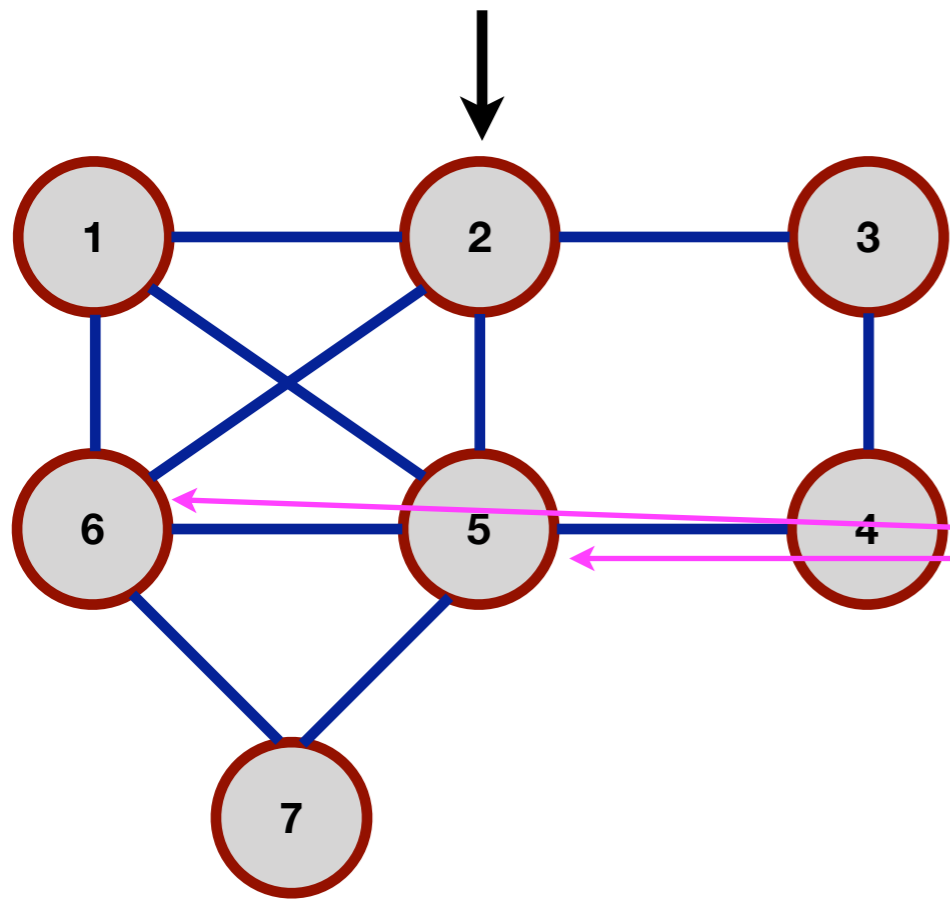
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7



Depth-First Search / Traversal

myGraph.DFS(vertex1)



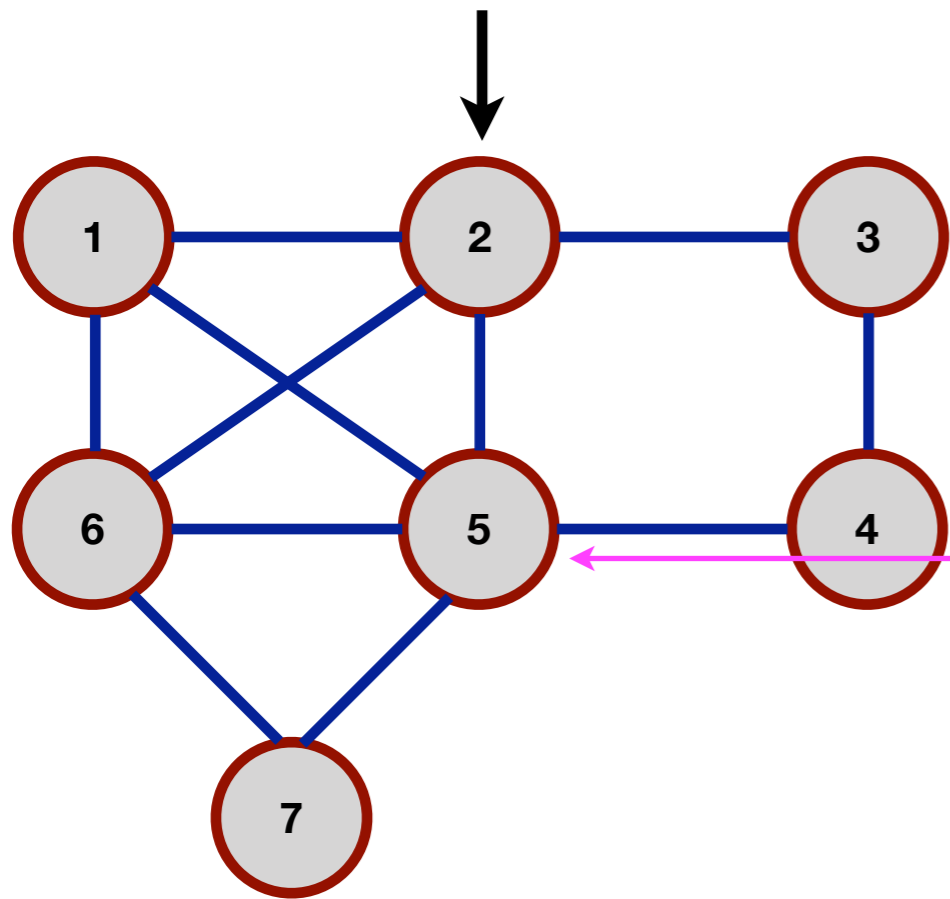
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



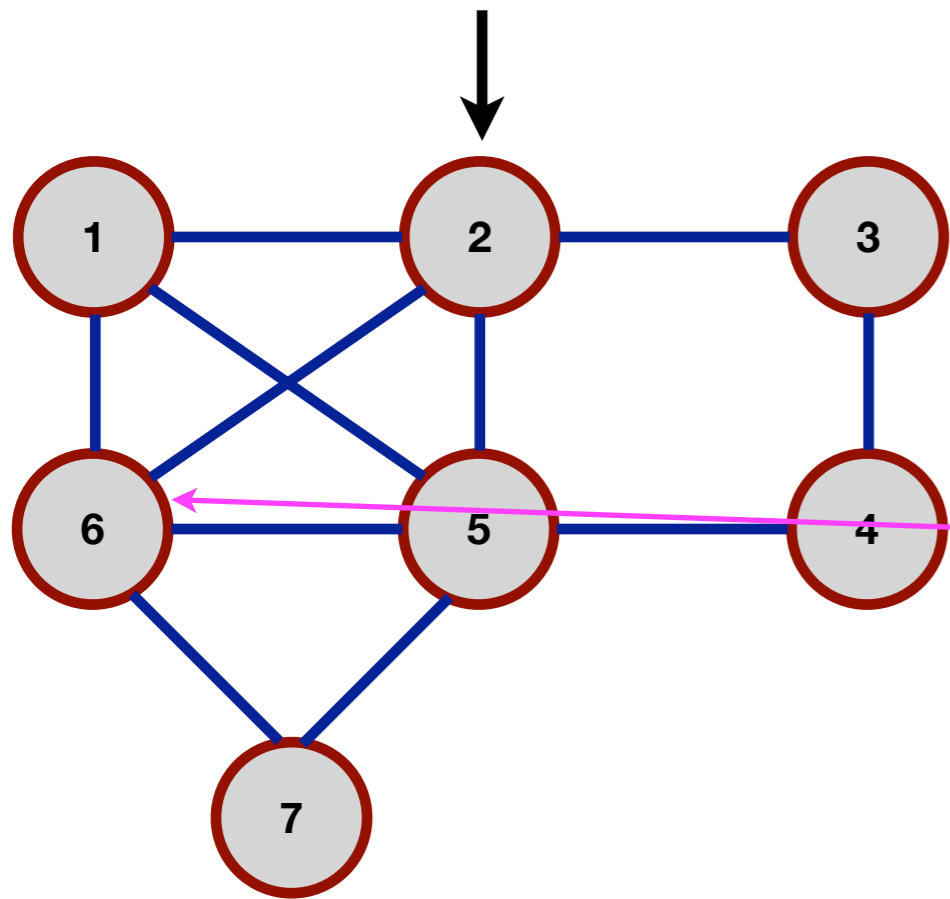
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



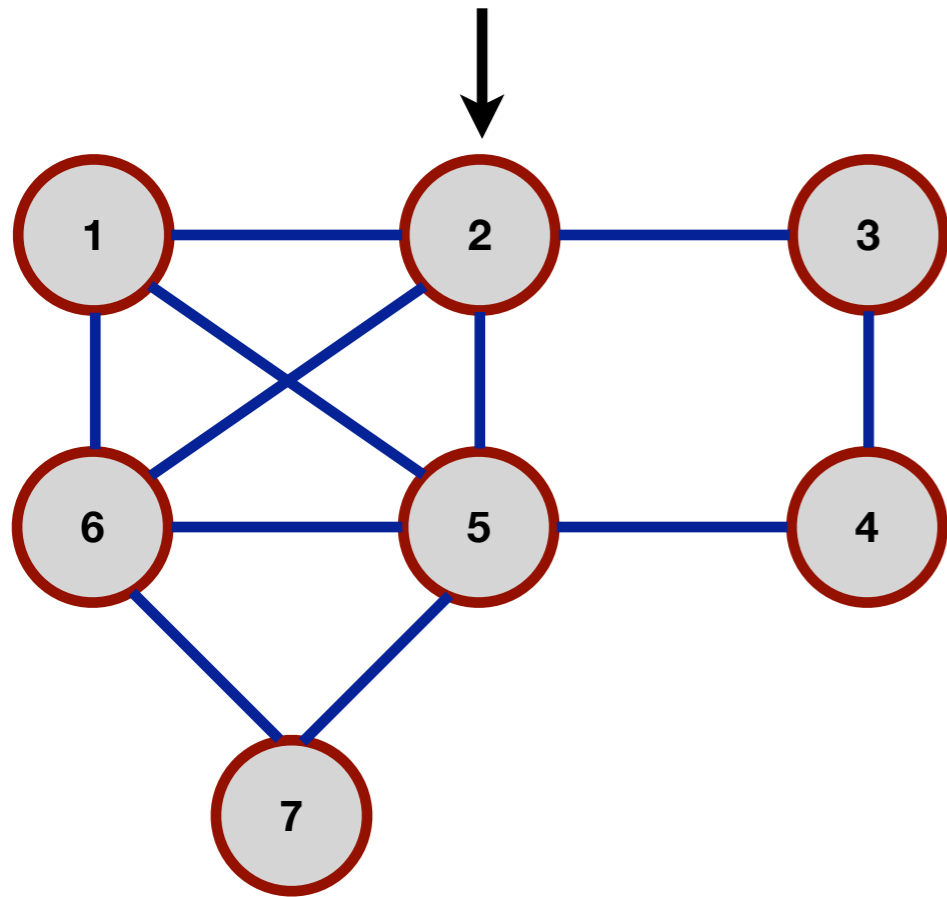
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



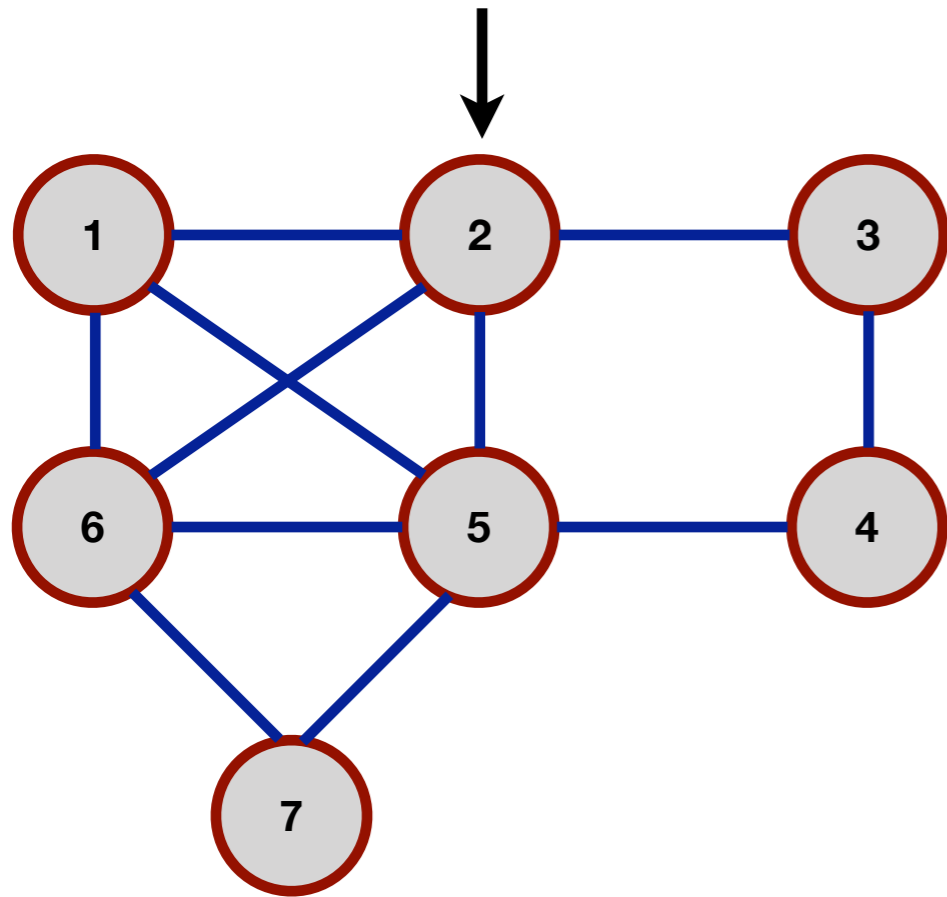
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



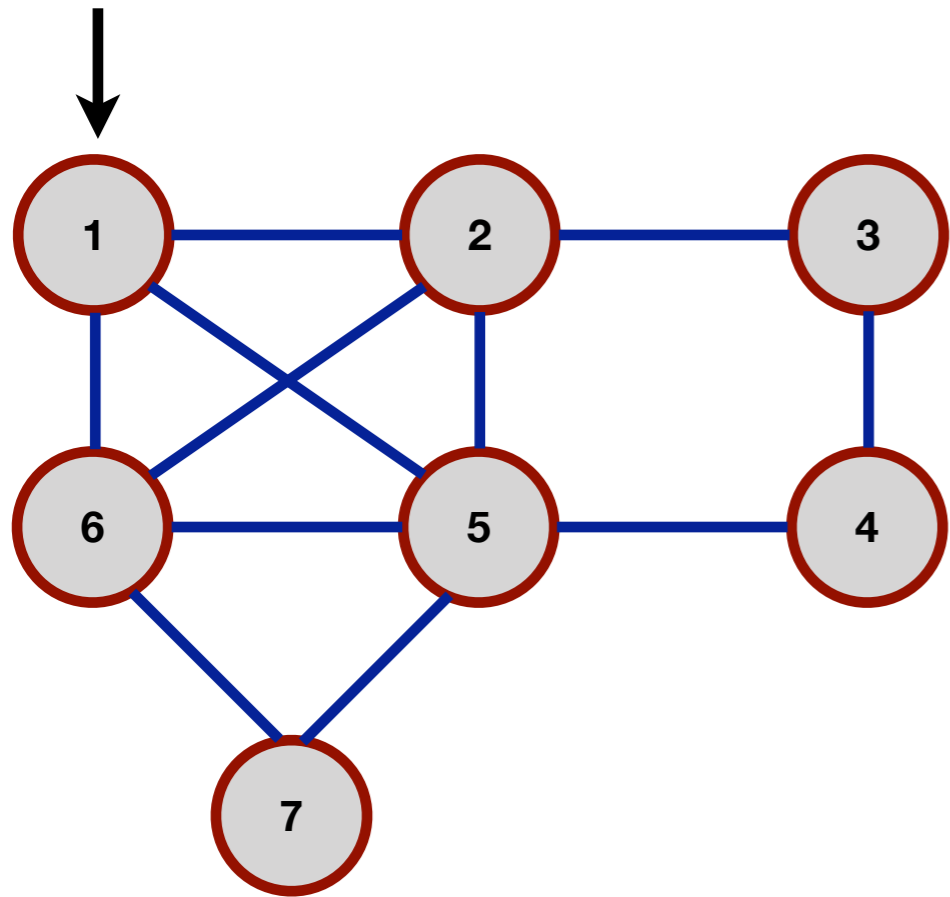
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

return address
Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



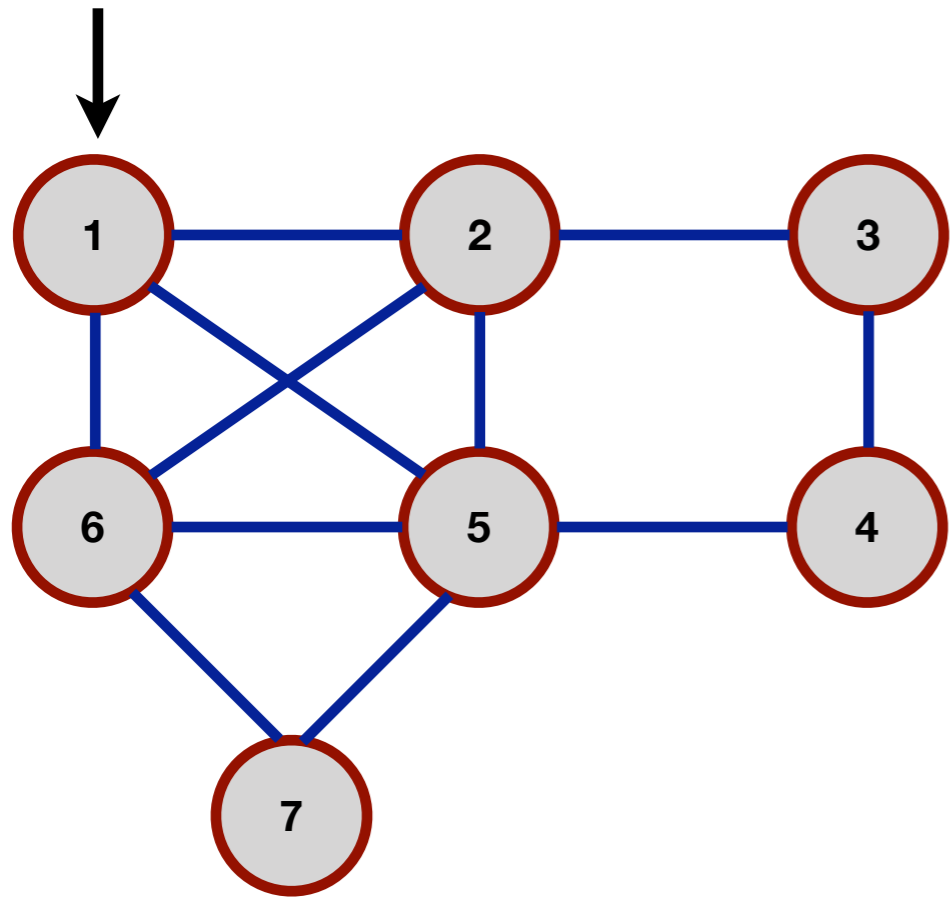
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



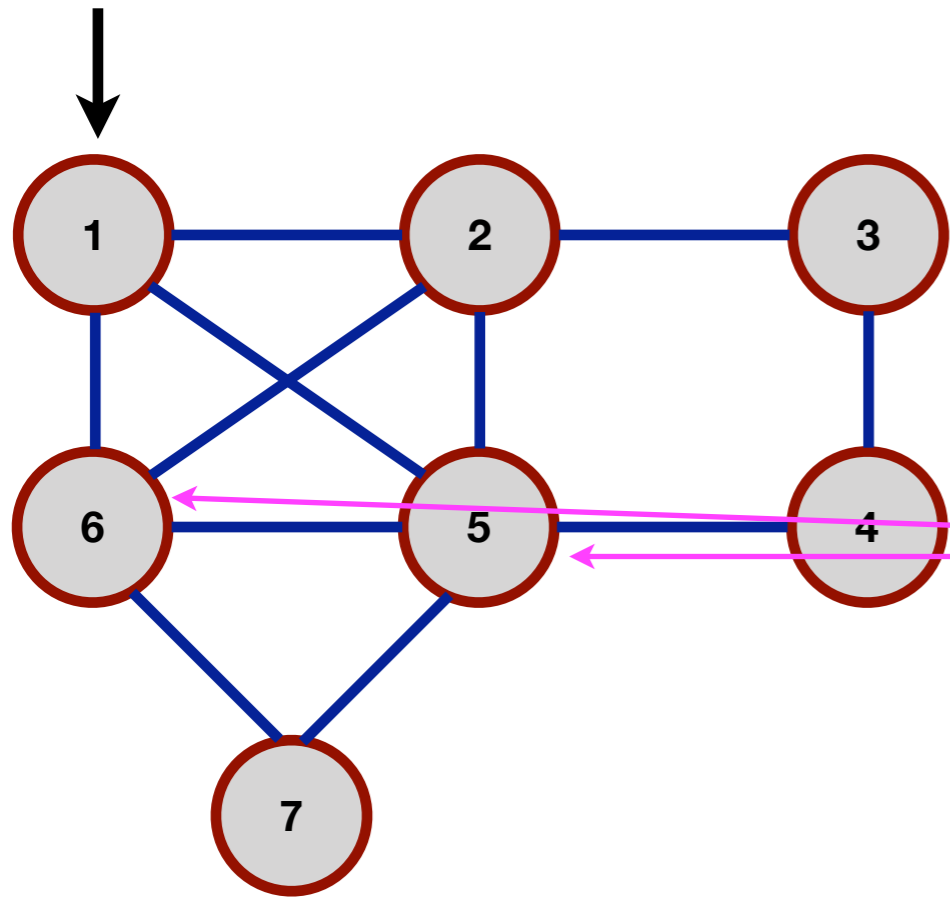
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



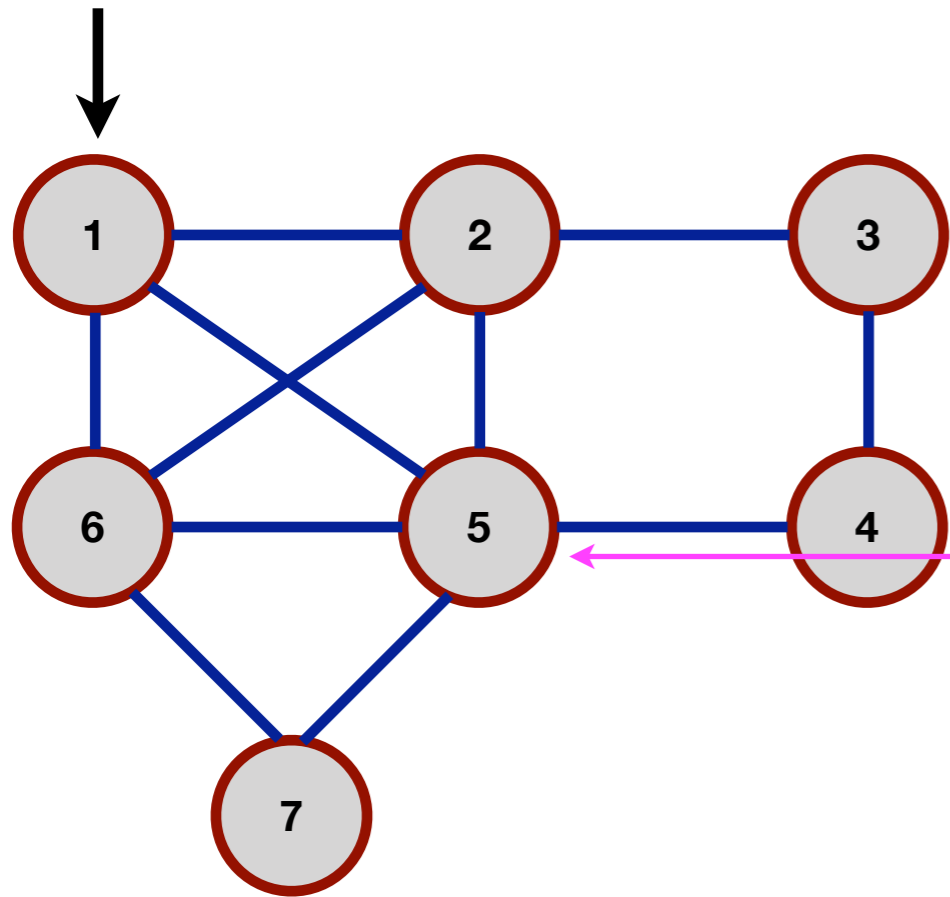
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



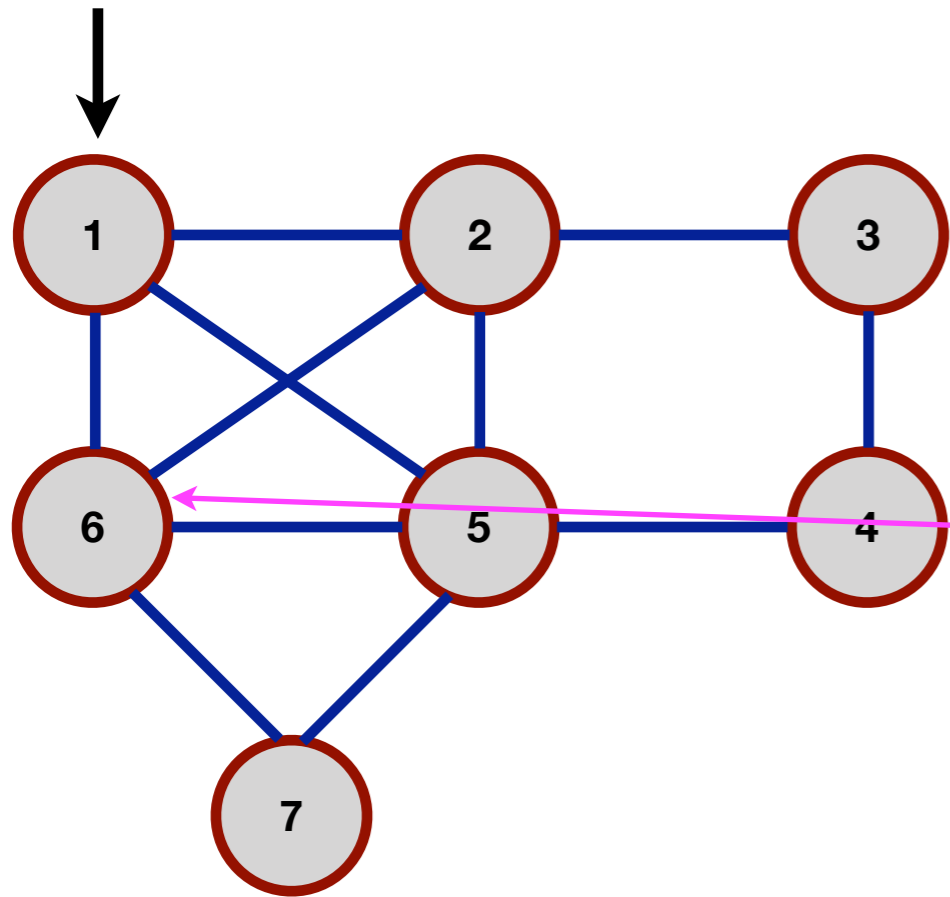
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



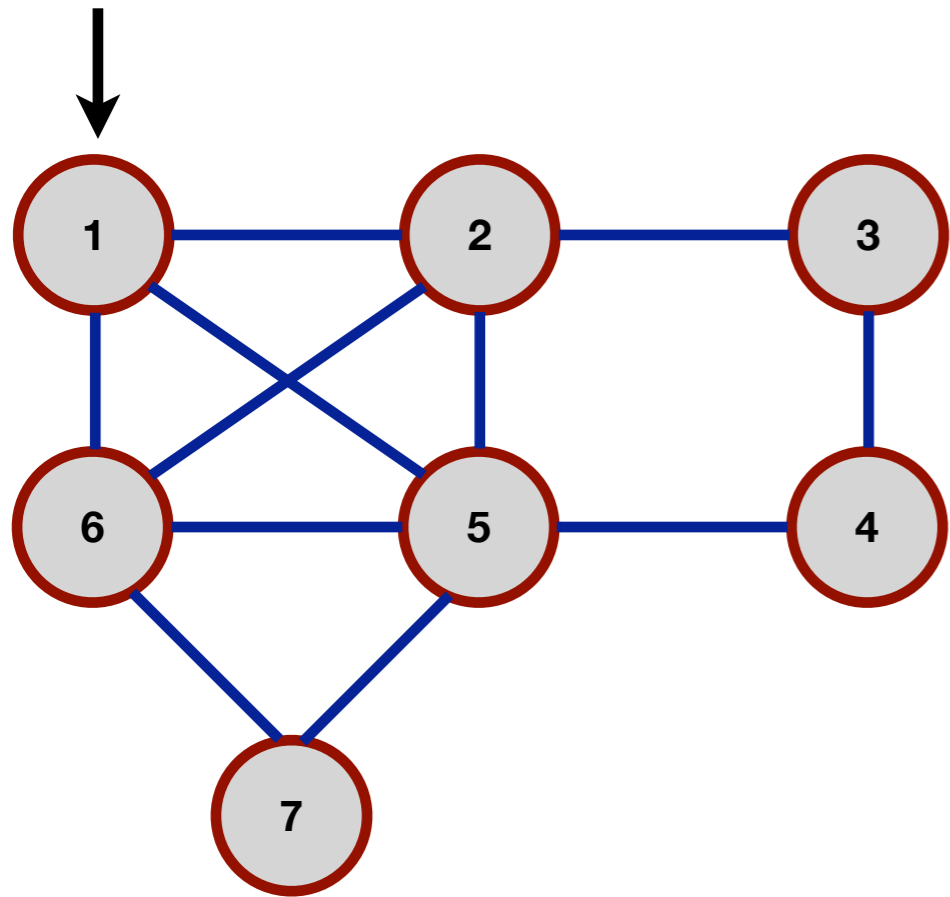
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



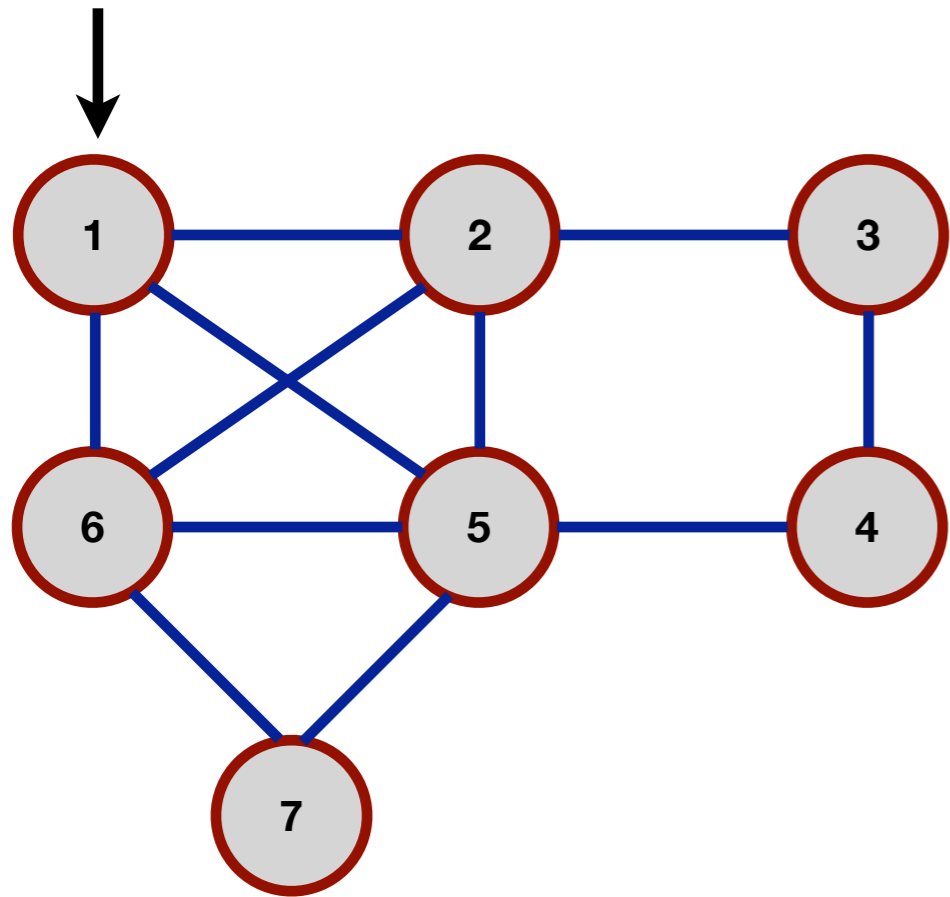
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



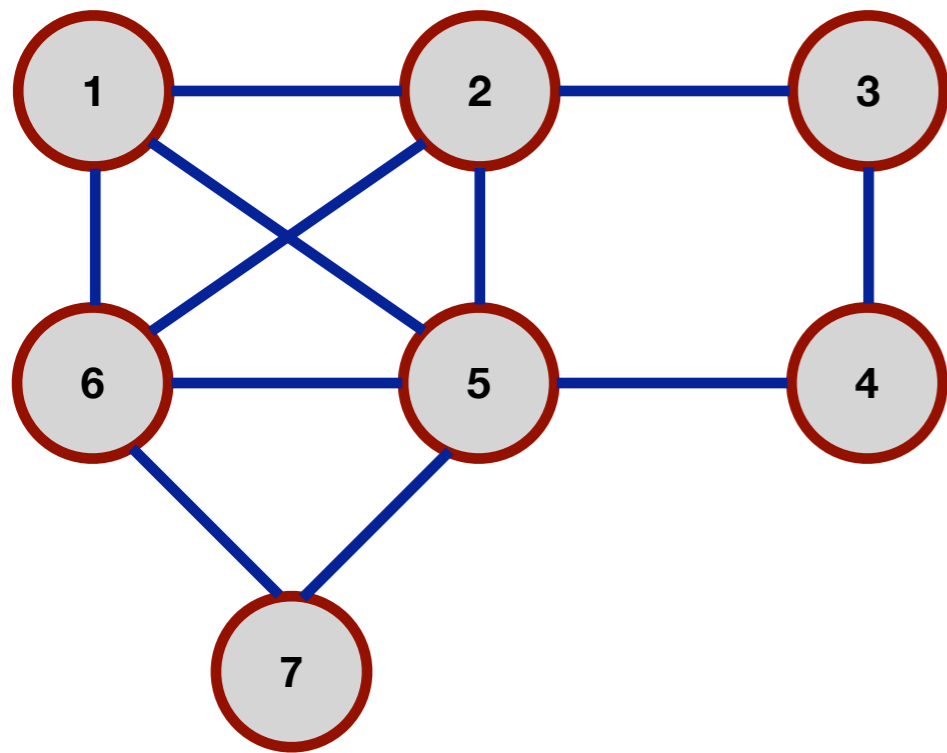
```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7

Runtime Stack

Depth-First Search / Traversal

myGraph.DFS(vertex1)



```
proc DFS(fromVertex v)
  if (not v.processed)
    print(v.id)
    v.processed := true
  endif
  for n in v.neighbors[]
    if (not n.processed)
      DFS(n)
    endif
  endfor
endproc
```

1 2 3 4 5 6 7