Chapter 14: Graphs

Shortest Path algorithms continues
Depth First Search

The DFS algorithm moves along one path as far as possible before backtracking and examining other paths off the earlier discovered vertices.

During the DFS execution, each vertex goes through three phases:

1. the vertex has not yet been discovered.

2. the vertex has been discovered, but the algorithm has not completed processing of all the vertices accessible from it.

3. we finished processing the vertex and all the vertices reachable from it.
Depth First Search

def(s):  
    for each vertex v in g:
        set v’s start time to 0
    t=0
    for each vertex v in g:
        if v’s start time is 0:
            dfs_traverse(g,v)

def_traverse(g,v):
    t += 1
    set v’s start time to t
    for each vertex u adjacent to v:
        if u’s start time is 0:
            set u’s parent to v
            dfs_traverse(g,u)
    t += 1
    set v’s end time to t
**Depth First Search**

`dfs(g):`

- **for each vertex** `v` **in** `g`:  
  - set `v`'s start time to 0

`t=0`

- **for each vertex** `v` **in** `g`:
  - **if** `v`'s start time is 0:
    - `dfs_traverse(g,v)`

`dfs_traverse(g,v):`

- `t += 1`
- set `v`'s start time to `t`
- **for each vertex** `u` **adjacent to** `v`:
  - **if** `u`'s start time is 0:
    - set `u`'s parent to `v`
    - `dfs_traverse(g,u)`
  - `t += 1`
- set `v`'s end time to `t`

---

<table>
<thead>
<tr>
<th></th>
<th>S</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>st</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>et</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

*CSI33 Data Structures*
**Depth First Search**

\[ \text{dfs}(g): \]
\[ \quad \text{for each vertex } v \text{ in } g: \]
\[ \quad \quad \text{set } v\text{'s start time to 0} \]
\[ \quad t=0 \]
\[ \quad \text{for each vertex } v \text{ in } g: \]
\[ \quad \quad \text{if } v\text{'s start time is 0:} \]
\[ \quad \quad \quad \text{dfs\_traverse}(g,v) \]

\[ \text{dfs\_traverse}(g,v): \]
\[ \quad t += 1 \]
\[ \quad \text{set } v\text{'s start time to } t \]
\[ \quad \text{for each vertex } u \text{ adjacent to } v: \]
\[ \quad \quad \text{if } u\text{'s start time is 0:} \]
\[ \quad \quad \quad \text{set } u\text{'s parent to } v \]
\[ \quad \quad \text{dfs\_traverse}(g,u) \]
\[ \quad t += 1 \]
\[ \quad \text{set } v\text{'s end time to } t \]

\[
\begin{array}{c|cccccc}
| & S & A & B & C & D \\
\hline
\text{par} & | & | & | & | & | \\
\hline
\text{st} & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\text{et} & | & | & | & | & | \\
\hline
\end{array}
\]

\[ t = 0 \]
**Depth First Search**

\[
dfs(g):
\]
\[
\text{for each vertex } v \text{ in } g:
\]
\[
\text{set } v \text{'s start time to } 0
\]
\[
t=0
\]
\[
\text{for each vertex } v \text{ in } g:
\]
\[
\text{if } v \text{'s start time is } 0:
\]
\[
dfs\_traverse(g,v)
\]

\[
dfs\_traverse(g,v):
\]
\[
t += 1
\]
\[
\text{set } v \text{'s start time to } t
\]
\[
\text{for each vertex } u \text{ adjacent to } v:
\]
\[
\text{if } u \text{'s start time is } 0:
\]
\[
\text{set } u \text{'s parent to } v
\]
\[
dfs\_traverse(g,u)
\]
\[
t += 1
\]
\[
\text{set } v \text{'s end time to } t
\]
**Depth First Search**

```python
dfs(g):
    for each vertex v in g:
        set v’s start time to 0
    t=0
    for each vertex v in g:
        if v’s start time is 0:
            dfs_traverse(g,v)

dfs_traverse(g,v):
    t += 1
    set v’s start time to t
    for each vertex u adjacent to v:
        if u’s start time is 0:
            set u’s parent to v
            dfs_traverse(g,u)
    t += 1
    set v’s end time to t
```

<table>
<thead>
<tr>
<th>v: S’s start time is 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>par</td>
</tr>
<tr>
<td>st</td>
</tr>
<tr>
<td>et</td>
</tr>
</tbody>
</table>

```

\[
t = 0
\]
Depth First Search

dfs(g):
  for each vertex v in g:
    set v’s start time to 0
  t = 0
  for each vertex v in g:
    if v’s start time is 0:
      dfs_traverse(g,v)

dfs_traverse(g,v):
  t += 1
  set v’s start time to t
  for each vertex u adjacent to v:
    if u’s start time is 0:
      set u’s parent to v
      dfs_traverse(g,u)
  t += 1
  set v’s end time to t

v:  S
    ------------------------
    | | S | A | B | C | D |
    ------------------------
    |par| | | | | |
    ------------------------
    | st| 1 | 0 | 0 | 0 | 0 |
    ------------------------
    | et|   |   |   |   |   |
    ------------------------
    t = 1
**Depth First Search**

**dfs(g):**
- for each vertex v in g:
  - set v’s start time to 0
- t = 0
- for each vertex v in g:
  - if v’s start time is 0:
    - dfs_traverse(g,v)

**dfs_traverse(g,v):**
- t += 1
- set v’s start time to t
- for each vertex u adjacent to v:
  - if u’s start time is 0:
    - set u’s parent to v
    - dfs_traverse(g,u)
- t += 1
- set v’s end time to t

---

<table>
<thead>
<tr>
<th>v:</th>
<th>S</th>
<th>u:</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>et</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

$\text{t} = 1$
**Depth First Search**

DFS$(g)$:
   
   for each vertex $v$ in $g$:
       set $v$'s start time to 0

   $t = 0$

   for each vertex $v$ in $g$:
       if $v$'s start time is 0:
           DFS$_{traverse}(g,v)$

DFS$_{traverse}(g,v)$:

   $t += 1$

   set $v$'s start time to $t$

   for each vertex $u$ adjacent to $v$:
       if $u$'s start time is 0:
           set $u$'s parent to $v$
           DFS$_{traverse}(g,u)$

   $t += 1$

   set $v$'s end time to $t$

---

<table>
<thead>
<tr>
<th>v</th>
<th>S</th>
<th>u</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
**Depth First Search**

**dfs(g):**
- for each vertex v in g:
  - set v’s start time to 0
- t=0
- for each vertex v in g:
  - if v’s start time is 0:
    - dfs_traverse(g,v)

**dfs_traverse(g,v):**
- t += 1
- set v’s start time to t
- for each vertex u adjacent to v:
  - if u’s start time is 0:
    - set u’s parent to v
    - dfs_traverse(g,u)
- t += 1
- set v’s end time to t

---

<table>
<thead>
<tr>
<th>v</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
</tr>
<tr>
<td>st</td>
<td>2</td>
</tr>
<tr>
<td>et</td>
<td></td>
</tr>
<tr>
<td>s</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td></td>
</tr>
<tr>
<td>d</td>
<td></td>
</tr>
</tbody>
</table>

\[ t = 2 \]
**Depth First Search**

**dfs(g):**

```python
    for each vertex v in g:
        set v’s start time to 0
    t=0
    for each vertex v in g:
        if v’s start time is 0:
            dfs_traverse(g,v)
```

**dfs_traverse(g,v):**

```python
t += 1
set v’s start time to t
for each vertex u adjacent to v:
    if u’s start time is 0:
        set u’s parent to v
        dfs_traverse(g,u)
t += 1
set v’s end time to t
```

---

<table>
<thead>
<tr>
<th>v</th>
<th>B</th>
<th>u:C</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
<td>S</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>et</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

---

\[ t = 2 \]
**Depth First Search**

defs(g):
    for each vertex v in g:
        set v’s start time to 0
    t=0
    for each vertex v in g:
        if v’s start time is 0:
            dfs_traverse(g,v)

defs_traverse(g,v):
    t += 1
    set v’s start time to t
    for each vertex u adjacent to v:
        if u’s start time is 0:
            set u’s parent to v
            dfs_traverse(g,u)
    t += 1
    set v’s end time to t

---

```
v:  B       u:C
-------------------------
|    | S | A | B | C | D |
-------------------------
|par|   | S | B |   |
-------------------------
|st  | 1 | 0 | 2 | 0 | 0 |
-------------------------
|et  |   |  |   |   |   |
-------------------------
t = 2
```
**Depth First Search**

```python
def dfs(g):
    for each vertex v in g:
        set v's start time to 0
    t=0
    for each vertex v in g:
        if v's start time is 0:
            dfs_traverse(g,v)

def dfs_traverse(g,v):
    t += 1
    set v's start time to t
    for each vertex u adjacent to v:
        if u's start time is 0:
            set u's parent to v
            dfs_traverse(g,u)
    t += 1
    set v's end time to t
```

<table>
<thead>
<tr>
<th>v:</th>
<th>B</th>
<th>u: C</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
<td>S A B C D</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>0 2 3 0</td>
</tr>
<tr>
<td>et</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ t = 3 \]
**Depth First Search**

\[ \text{dfs}(g): \]
   \[ \text{for each vertex } v \text{ in } g: \]
      \[ \text{set } v \text{'s start time to } 0 \]
   \[ t = 0 \]
   \[ \text{for each vertex } v \text{ in } g: \]
      \[ \text{if } v \text{'s start time is } 0: \]
         \[ \text{dfs}_\text{traverse}(g,v) \]

\[ \text{dfs}_\text{traverse}(g,v): \]
   \[ t += 1 \]
   \[ \text{set } v \text{'s start time to } t \]
   \[ \text{for each vertex } u \text{ adjacent to } v: \]
      \[ \text{if } u \text{'s start time is } 0: \]
         \[ \text{set } u \text{'s parent to } v \]
         \[ \text{dfs}_\text{traverse}(g,u) \]
   \[ t += 1 \]
   \[ \text{set } v \text{'s end time to } t \]
**Depth First Search**

\[
\text{dfs}(g):
\]
\[
\text{for each vertex } v \text{ in } g:
\quad \text{set } v\text{'s start time to 0}
\]
\[
t=0
\]
\[
\text{for each vertex } v \text{ in } g:
\quad \text{if } v\text{'s start time is 0:}
\quad \quad \text{dfs\_traverse}(g,v)
\]

\[
\text{dfs\_traverse}(g,v):
\]
\[
t += 1
\]
\[
\text{set } v\text{'s start time to } t
\]
\[
\text{for each vertex } u \text{ adjacent to } v:
\quad \text{if } u\text{'s start time is 0:}
\quad \quad \text{set } u\text{'s parent to } v
\quad \quad \text{dfs\_traverse}(g,u)
\]
\[
t += 1
\]
\[
\text{set } v\text{'s end time to } t
\]

\[
\begin{array}{c|c|c|c|c|c|c}
\hline
v & B & & & C \\
\hline
\hline
par & & & S & B & C & D \\
\hline
st & 1 & 0 & 2 & 3 & 0 \\
\hline
et & & & & 4 & & \\
\hline
\end{array}
\]

\[
t = 4
\]
**Depth First Search**

\[
\text{dfs}(g):
\]
\[
\text{for each vertex } v \text{ in } g:\n\]
\[
\text{set } v\text{’s start time to 0}
\]
\[
 t=0
\]
\[
\text{for each vertex } v \text{ in } g:
\]
\[
\text{if } v\text{’s start time is 0:}
\]
\[
\text{dfs_traverse}(g,v)
\]

\[
\text{dfs_traverse}(g,v):
\]
\[
 t += 1
\]
\[
\text{set } v\text{’s start time to } t
\]
\[
\text{for each vertex } u \text{ adjacent to } v:
\]
\[
\text{if } u\text{’s start time is 0:}
\]
\[
\text{set } u\text{’s parent to } v
\]
\[
\text{dfs_traverse}(g,u)
\]
\[
 t += 1
\]
\[
\text{set } v\text{’s end time to } t
\]

---

<table>
<thead>
<tr>
<th>v</th>
<th>B</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td>S</td>
<td>B</td>
</tr>
<tr>
<td>st</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>et</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
t = 4
\]
Depth First Search

dfs(g):
    for each vertex v in g:
        set v’s start time to 0
    t=0
    for each vertex v in g:
        if v’s start time is 0:
            dfs_traverse(g,v)

dfs_traverse(g,v):
    t += 1
    set v’s start time to t
    for each vertex u adjacent
to v:
        if u’s start time is 0:
            set u’s parent to v
            dfs_traverse(g,u)
    t += 1
    set v’s end time to t

<table>
<thead>
<tr>
<th>v</th>
<th>B</th>
<th>u:D</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td></td>
<td>S</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>et</td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

\[ t = 4 \]
**Depth First Search**

\[ \text{dfs}(g): \]
\[
\begin{align*}
\text{for each vertex } v \text{ in } g: \\
& \text{set } v's \text{ start time to } 0 \\
\text{t=0} \\
\text{for each vertex } v \text{ in } g: \\
& \text{if } v's \text{ start time is } 0: \\
& \quad \text{dfs\_traverse}(g,v) \\
\end{align*}
\]

\[ \text{dfs\_traverse}(g,v): \]
\[
\begin{align*}
& \text{t += 1} \\
& \text{set } v's \text{ start time to } t \\
& \text{for each vertex } u \text{ adjacent to } v: \\
& \quad \text{if } u's \text{ start time is } 0: \\
& \quad \quad \text{set } u's \text{ parent to } v \\
& \quad \quad \text{dfs\_traverse}(g,u) \\
& \text{t += 1} \\
& \text{set } v's \text{ end time to } t \\
\end{align*}
\]

\[
\begin{array}{cccccccc}
\text{v:} & D \\
\hline
\text{par} & S & A & B & C & D \\
\hline
\text{st} & 1 & 0 & 2 & 3 & 5 \\
\hline
\text{et} & & & & 4 & \\
\end{array}
\]

\[ t = 5 \]
Depth First Search

dfs(g):
   for each vertex v in g:
      set v’s start time to 0
   t=0
   for each vertex v in g:
      if v’s start time is 0:
         dfs_traverse(g,v)

dfs_traverse(g,v):
   t += 1
   set v’s start time to t
   for each vertex u adjacent to v:
      if u’s start time is 0:
         set u’s parent to v
         dfs_traverse(g,u)
   t += 1
   set v’s end time to t
**Depth First Search**

\[ \text{dfs}(g): \]
\[ \text{for each vertex } v \text{ in } g: \]
\[ \quad \text{set } v's \text{ start time to 0} \]
\[ t=0 \]
\[ \text{for each vertex } v \text{ in } g: \]
\[ \quad \text{if } v's \text{ start time is 0:} \]
\[ \quad \quad \text{dfs}_\text{traverse}(g,v) \]

\[ \text{dfs}_\text{traverse}(g,v): \]
\[ t += 1 \]
\[ \text{set } v's \text{ start time to } t \]
\[ \text{for each vertex } u \text{ adjacent to } v: \]
\[ \quad \text{if } u's \text{ start time is 0:} \]
\[ \quad \quad \text{set } u's \text{ parent to } v \]
\[ \quad \quad \text{dfs}_\text{traverse}(g,u) \]
\[ t += 1 \]
\[ \text{set } v's \text{ end time to } t \]

<table>
<thead>
<tr>
<th>v:</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td>D</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
</tr>
<tr>
<td>et</td>
<td>4</td>
</tr>
<tr>
<td>t</td>
<td>6</td>
</tr>
</tbody>
</table>

**Diagram Description**

- **Vertices:** A, B, C, D
- **Edges:**
  - S-A
  - A-B
  - A-D
  - B-C

**Table:**

- **Columns:** S, A, B, C, D
- **Rows:**
  - **par:** D, S, B, B
  - **st:** 1, 6, 2, 3
  - **et:** 4
  - **t:** 6

**Graph:**

A directed graph with vertices S, A, B, C, D and edges S-A, A-B, A-D, B-C.
Depth First Search

dfs(g):
   for each vertex v in g:
      set v’s start time to 0
   t=0
   for each vertex v in g:
      if v’s start time is 0:
         dfs_traverse(g,v)

dfs_traverse(g,v):
   t += 1
   set v’s start time to t
   for each vertex u adjacent to v:
      if u’s start time is 0:
         set u’s parent to v
         dfs_traverse(g,u)
   t += 1
   set v’s end time to t

v:  A

<table>
<thead>
<tr>
<th>par</th>
<th>D</th>
<th>S</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>st</td>
<td>1</td>
<td>6</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>et</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ t = 6 \]
**Depth First Search**

**dfs(g):**

for each vertex v in g:
    set v’s start time to 0

\[ t = 0 \]

for each vertex v in g:
    if v’s start time is 0:
        dfs_traverse(g,v)

**dfs_traverse(g,v):**

\[ t \mapsto 1 \]

set v’s start time to t

for each vertex u adjacent to v:
    if u’s start time is 0:
        set u’s parent to v
    dfs_traverse(g,u)

\[ t \mapsto 1 \]

set v’s end time to t

<table>
<thead>
<tr>
<th>v</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td>D</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
</tr>
<tr>
<td>et</td>
<td>7</td>
</tr>
</tbody>
</table>

\[ t = 7 \]
**Depth First Search**

\[
\text{dfs}(g): \\
\quad \text{for each vertex } v \text{ in } g: \\
\quad \quad \text{set } v\text{'s start time to } 0 \\
\quad t=0 \\
\quad \text{for each vertex } v \text{ in } g: \\
\quad \quad \text{if } v\text{'s start time is } 0: \\
\quad \quad \quad \text{dfs\_traverse}(g,v) \\
\]

\[
\text{dfs\_traverse}(g,v): \\
\quad t += 1 \\
\quad \text{set } v\text{'s start time to } t \\
\quad \text{for each vertex } u \text{ adjacent to } v: \\
\quad \quad \text{if } u\text{'s start time is } 0: \\
\quad \quad \quad \text{set } u\text{'s parent to } v \\
\quad \quad \quad \text{dfs\_traverse}(g,u) \\
\quad t += 1 \\
\quad \text{set } v\text{'s end time to } t \\
\]

<table>
<thead>
<tr>
<th>v</th>
<th>D</th>
<th>u:C</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td>D</td>
<td>S</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>et</td>
<td>7</td>
<td>4</td>
</tr>
</tbody>
</table>

\[
t = 6
\]
Depth First Search

\[ \text{dfs}(g) : \]
\begin{verbatim}
  for each vertex \( v \) in \( g \):
    set \( v \)'s start time to 0
  \( t = 0 \)
  for each vertex \( v \) in \( g \):
    if \( v \)'s start time is 0:
      \text{dfs_traverse}(g,v)
\end{verbatim}

\[ \text{dfs_traverse}(g,v) : \]
\begin{verbatim}
  \( t += 1 \)
  set \( v \)'s start time to \( t \)
  for each vertex \( u \) adjacent to \( v \):
    if \( u \)'s start time is 0:
      set \( u \)'s parent to \( v \)
      \text{dfs_traverse}(g,u)
  \( t += 1 \)
  set \( v \)'s end time to \( t \)
\end{verbatim}

\[ \begin{array}{c|c|c|c|c|c}
  \text{v} & D & u:C \\
  \hline
  \text{par} & D & S & A & B & C & D \\
  \text{st} & 1 & 6 & 2 & 3 & 5 \\
  \text{et} & | & 7 | & 4 | & 8 |
\end{array} \]
**Depth First Search**

```plaintext
dfs(g):
    for each vertex v in g:
        set v’s start time to 0
    t=0
    for each vertex v in g:
        if v’s start time is 0:
            dfs_traverse(g,v)

dfs_traverse(g,v):
    t += 1
    set v’s start time to t
    for each vertex u adjacent to v:
        if u’s start time is 0:
            set u’s parent to v
            dfs_traverse(g,u)
    t += 1
    set v’s end time to t
```

<table>
<thead>
<tr>
<th>v</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td>D</td>
</tr>
<tr>
<td>st</td>
<td>1</td>
</tr>
<tr>
<td>et</td>
<td>7</td>
</tr>
</tbody>
</table>

\[ t = 6 \]
Depth First Search

\texttt{dfs}(g): 
  \hspace{1em} \textbf{for} each vertex \( v \) in \( g \):
  \hspace{2em} set \( v \)'s start time to 0
  \( t = 0 \)
  \hspace{1em} \textbf{for} each vertex \( v \) in \( g \):
    \hspace{2em} \textbf{if} \( v \)'s start time is 0:
      \hspace{3em} \texttt{dfs\_traverse}(g,v)

\texttt{dfs\_traverse}(g,v):
  \hspace{1em} \( t += 1 \)
  \hspace{2em} set \( v \)'s start time to \( t \)
  \hspace{1em} \textbf{for} each vertex \( u \) adjacent to \( v \):
    \hspace{2em} \textbf{if} \( u \)'s start time is 0:
      \hspace{3em} set \( u \)'s parent to \( v \)
      \hspace{3em} \texttt{dfs\_traverse}(g,u)
  \hspace{1em} \( t += 1 \)
  \hspace{2em} set \( v \)'s end time to \( t \)

\begin{tabular}{c|c|c|c|c|c|c}
\hline
\textbf{v} & B & \\
\hline
\textbf{par} & D & S & B & B & \\
\hline
\textbf{st} & 1 & 6 & 2 & 3 & 5 & \\
\hline
\textbf{et} & 7 & 9 & 4 & 8 & \\
\hline
\end{tabular}
\hspace{1em} \( t = 9 \)
**Depth First Search**

\[
\text{dfs}(g): \\
\text{for each vertex } v \text{ in } g: \\
\quad \text{set } v\text{’s start time to } 0 \\
\text{t=0} \\
\text{for each vertex } v \text{ in } g: \\
\quad \text{if } v\text{’s start time is } 0: \\
\quad \quad \text{dfs_traverse}(g,v) \\
\]

\[
\text{dfs_traverse}(g,v): \\
\quad \text{t += 1} \\
\quad \text{set } v\text{’s start time to } t \\
\quad \text{for each vertex } u \text{ adjacent to } v: \\
\quad \quad \text{if } u\text{’s start time is } 0: \\
\quad \quad \quad \text{set } u\text{’s parent to } v \\
\quad \quad \text{dfs_traverse}(g,u) \\
\quad \text{t += 1} \\
\quad \text{set } v\text{’s end time to } t \\
\]

\[
\begin{array}{c|c|c|c|c|c|c}
  v: & S & A & B & C & D \\
  \hline
  \text{par:} & D & S & B & B & \\
  \hline
  \text{st:} & 1 & 6 & 2 & 3 & 5 \\
  \hline
  \text{et:} & 7 & 9 & 4 & 8 & \\
  \hline
  \text{t} & 9 &
\end{array}
\]
**Depth First Search**

\[\text{dfs}(g):\]
\[
\quad \text{for each vertex } v \text{ in } g:\n\quad \quad \text{set } v\text{'s start time to 0}
\]
\[
\quad t=0
\]
\[
\quad \text{for each vertex } v \text{ in } g:\n\quad \quad \text{if } v\text{'s start time is 0:}
\quad \quad \quad \text{dfs\_traverse}(g,v)
\]

\[\text{dfs\_traverse}(g,v):\]
\[
\quad t += 1
\]
\[
\quad \text{set } v\text{'s start time to } t
\]
\[
\quad \text{for each vertex } u \text{ adjacent to } v:\n\quad \quad \text{if } u\text{'s start time is 0:}
\quad \quad \quad \text{set } u\text{'s parent to } v
\quad \quad \quad \text{dfs\_traverse}(g,u)
\]
\[
\quad t += 1
\]
\[
\quad \text{set } v\text{'s end time to } t
\]

---

**v:**  

<table>
<thead>
<tr>
<th>v</th>
<th>S</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>par</td>
<td>D</td>
<td>S</td>
<td>B</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>st</td>
<td>1</td>
<td>6</td>
<td>2</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>et</td>
<td>10</td>
<td>7</td>
<td>9</td>
<td>4</td>
<td>8</td>
</tr>
</tbody>
</table>

\[t = 10\]
**Depth First Search**

\[ \text{dfs}(g): \]
\[
\text{for each vertex } v \text{ in } g: \\
\quad \text{set } v\text{'s start time to 0}
\]
\[
t=0
\]
\[
\text{for each vertex } v \text{ in } g: \\
\quad \text{if } v\text{'s start time is 0:} \\
\quad \quad \text{dfs_traverse}(g,v)
\]

\[ \text{dfs_traverse}(g,v): \]
\[
\quad t += 1
\]
\[
\text{set } v\text{'s start time to } t
\]
\[
\text{for each vertex } u \text{ adjacent to } v:
\quad \text{if } u\text{'s start time is 0:} \\
\quad \quad \text{set } u\text{'s parent to } v \\
\quad \quad \text{dfs_traverse}(g,u)
\]
\[
\quad t += 1
\]
\[
\text{set } v\text{'s end time to } t
\]

<table>
<thead>
<tr>
<th>v: A,B,C,D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>------------------------</td>
</tr>
</tbody>
</table>

\[ t = 10 \]
**Depth First Search**

**Running time analysis**

The `dfs` function processes each vertex in a constant number of times.

The `dfs_traverse` function processes each edge once and performs a constant number of operations as it processes each edge.

Hence, the overall time is $\Theta(V + E)$. 