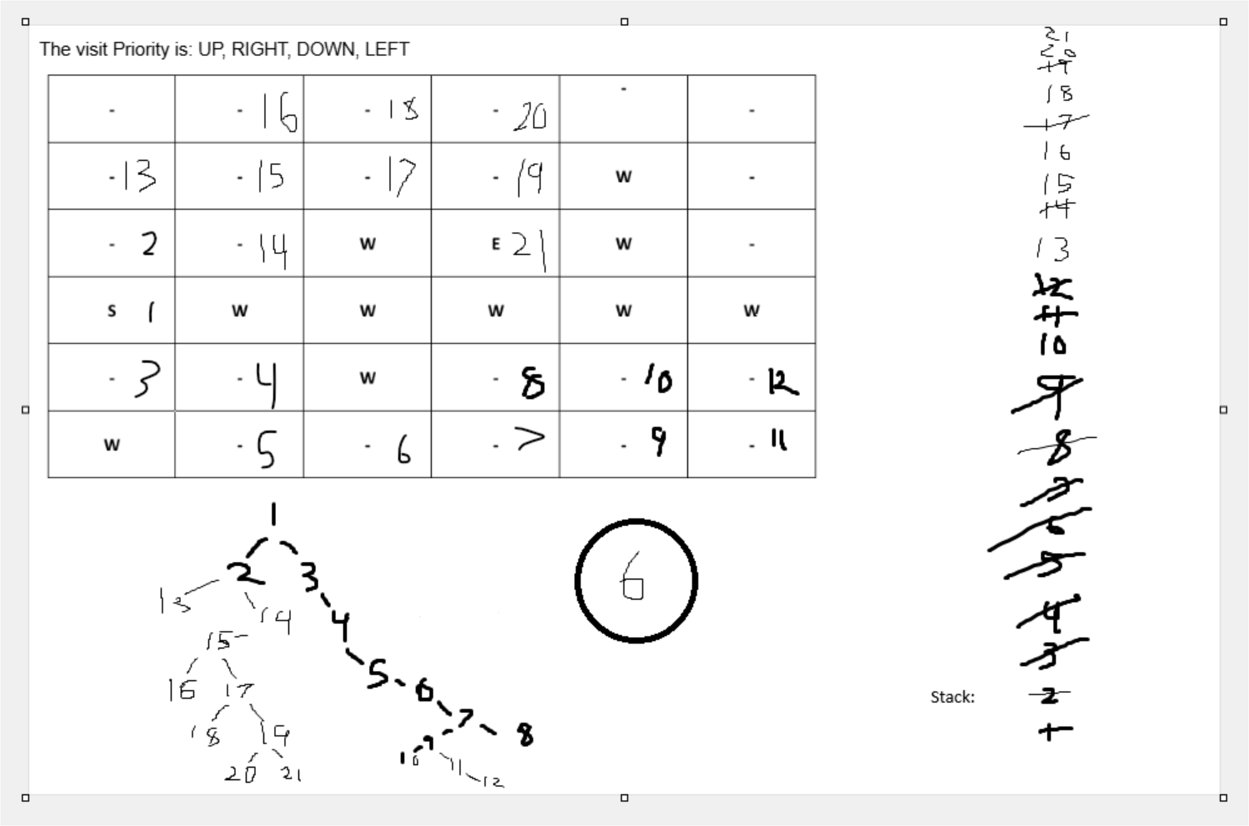
Name:___________________

Solve the following Maze using a DFS.

1. Create two areas below the grid. One for showing your stack and the other for a visitation graph.
2. Write a 1 on the grid location with ‘S’, push 1 to the stack and add a node with the text 1 to the visitation graph.
3. If the stack is empty go to step 8, otherwise go to step 4
4. Pop a number from the stack (referred to as **popped**)
5. If **popped** is the number in the ‘E’ cell go to step 8, otherwise go to step 6
6. For each cell that is not numbered and is both orthogonally adjacent and not a’W’. (**Access cells in order based on the visit priority listed above each graph)**
    a. Number the cell with the next available ascending number, referred to as **neighbor**.
    b. Push **neighbor** to the stack
    c. Create a node in the visitation graph with **neighbor**
    d. Connect the **neighbor** in the visitation graph to **popped**
7. Go to step 3
8. Write down the number of edges it takes to reach the number in the ‘E’ cell from the number in the ‘S’ cell in the visitation graph.

The visit Priority is: UP, RIGHT, DOWN, LEFT



The visit Priority is: UP, RIGHT, DOWN, LEFT

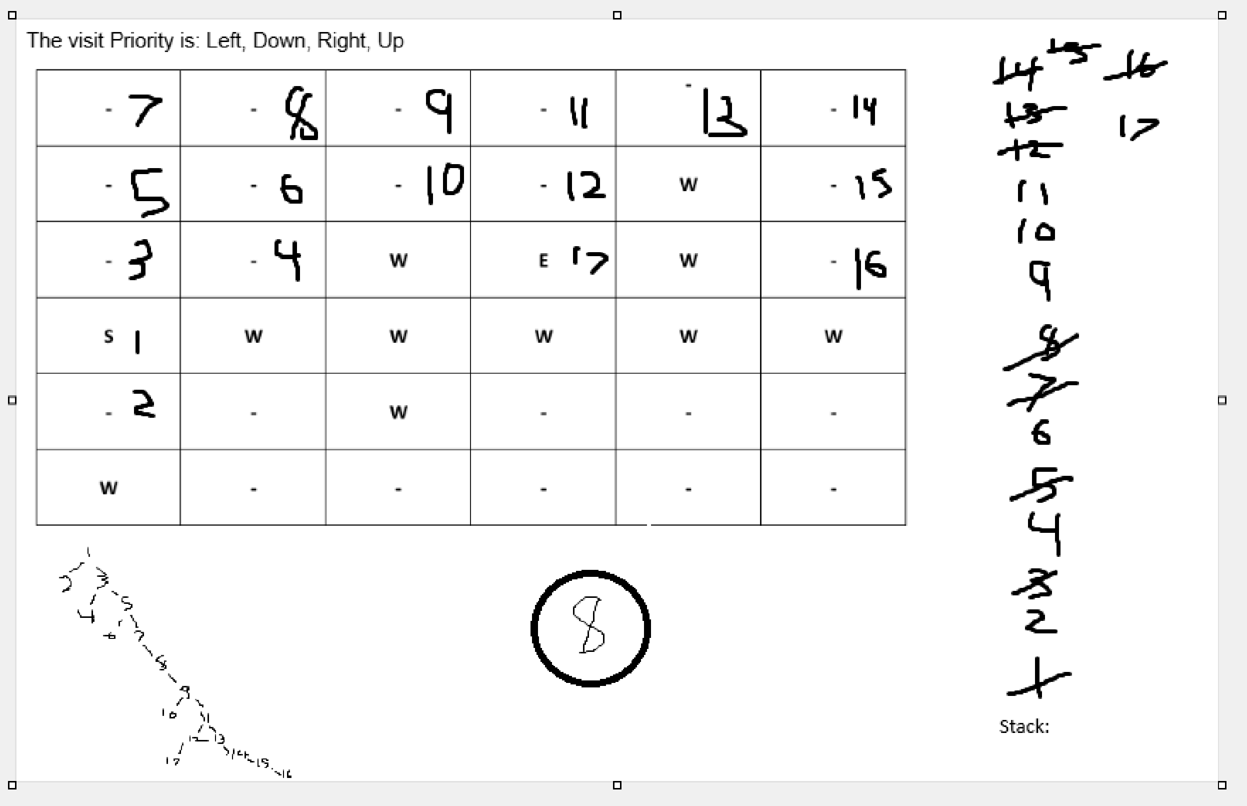
-	- 16	- 18	- 20	-	-
- 13	- 15	- 17	- 19	w	-
- 2	- 14	w	E 21	w	-
s 1	w	w	w	w	w
- 3	- 4	w	- 8	- 10	- 12
w	- 5	- 6	- 7	- 9	- 11

6

21
20
~~19~~
18
~~17~~
16
15
~~14~~
13
~~12~~
~~11~~
10
~~9~~
~~8~~
~~7~~
~~6~~
~~5~~
~~4~~
~~3~~
Stack: ~~2~~
~~1~~

Name:___________________

The visit Priority is: Left, Down, Right, Up



The visit Priority is: Left, Down, Right, Up

- 7	- 8	- 9	- 11	- 13	- 14
- 5	- 6	- 10	- 12	W	- 15
- 3	- 4	W	E 17	W	- 16
S 1	W	W	W	W	W
- 2	-	W	-	-	-
W	-	-	-	-	-

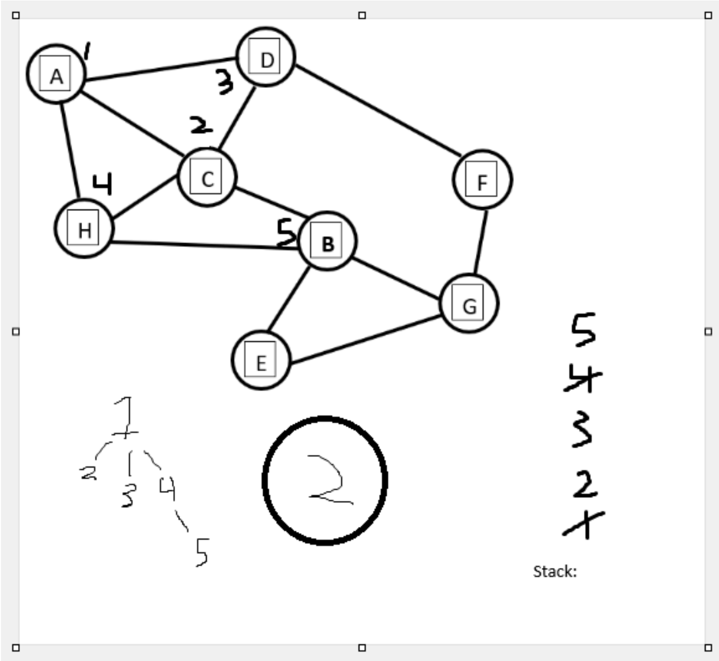
~~14~~ ~~15~~ ~~16~~
~~13~~ 17
~~12~~
11
10
9
~~8~~
~~7~~
6
~~5~~
4
~~3~~
2
~~1~~
Stack:

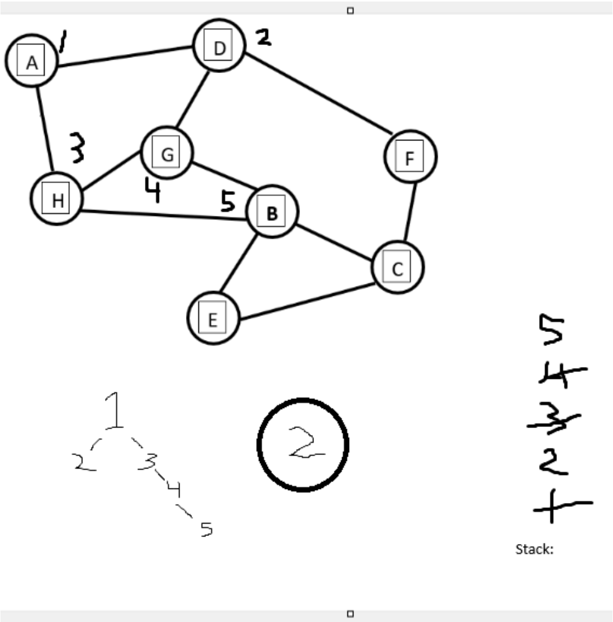
8

Name:____________________

Solve the following Maze using a DFS.

1. Create two areas below the Graph. One for showing your stack and the other for a visitation graph.
2. Write a 1 near the graph node with the label ‘A’, push 1 to the stack and add a node with the text 1 to the visitation graph.
3. If the stack is empty go to step 8, otherwise go to step 4
4. Pop a number from the stack (referred to as **popped**)
5. If **popped** is the number next to the ‘C’ node go to step 8, otherwise go to step 6
6. For each node that is not numbered and is adjacent **popped**. (**Access nodes in alphabetical order)**
    a. Write a number next to the node using the next available ascending number, referred to as **neighbor**.
    b. Push **neighbor** to the stack
    c. Create a node in the visitation graph with **neighbor**
    d. Connect the **neighbor** in the visitation graph to **popped**
7. Go to step 3
8. Write down the number of edges it takes to reach the number next to the ‘B’ node from the number next to the ‘A’ node in the visitation graph.





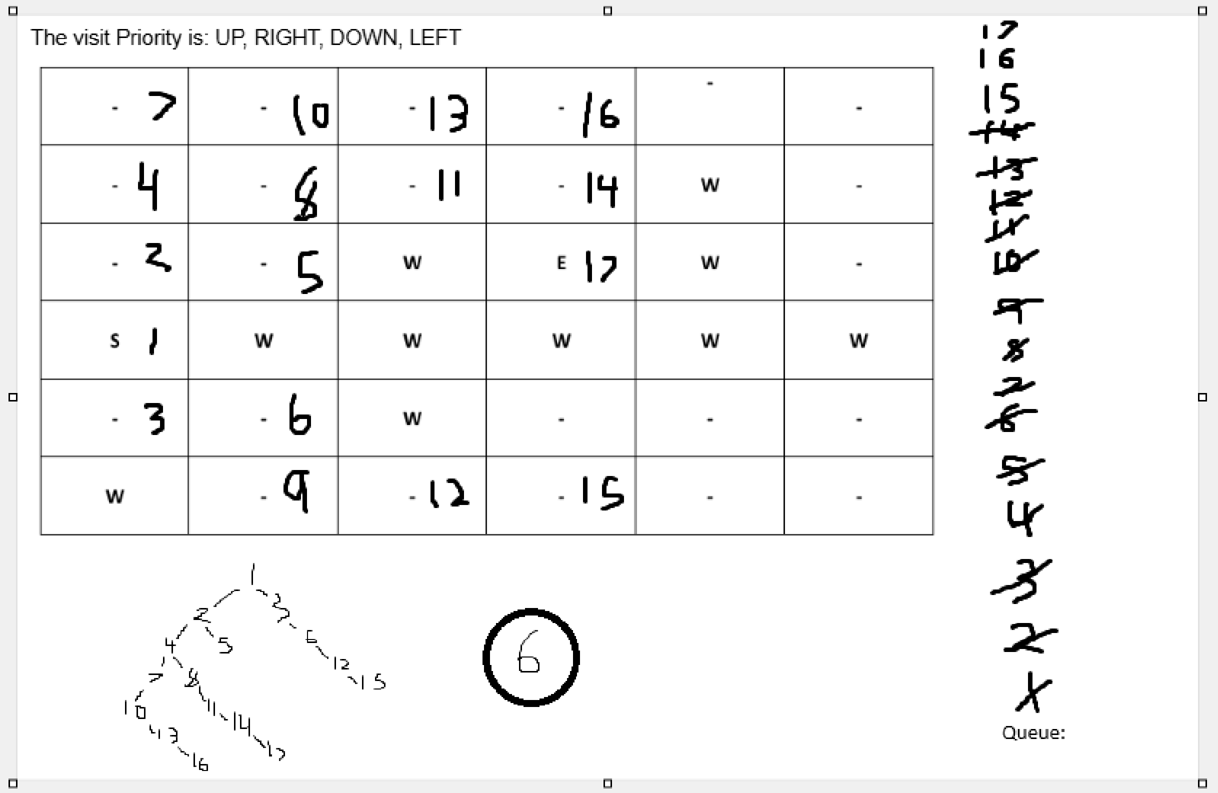
Name:___________________

Solve the following Maze using a BFS.

1. Create two areas below the grid. One for showing your queue and the other for a visitation graph.
2. Write a 1 on the grid location with ‘S’, offer 1 to the queue and add a node with the text 1 to the visitation graph.
3. If the queue is empty go to step 8, otherwise go to step 4
4. poll a number from the queue (referred to as **removed**)
5. If **removed** is the number in the ‘E’ cell go to step 8, otherwise go to step 6
6. For each cell that is not numbered and is both orthogonally adjacent and not a’W’. (**Access cells in order based on the visit priority listed above each graph)**
    a. Number the cell with the next available ascending number, referred to as **neighbor**.
    b. Offer **neighbor** to the queue
    c. Create a node in the visitation graph with **neighbor**
    d. Connect the **neighbor** in the visitation graph to **removed**
7. Go to step 3
8. Write down the number of edges it takes to reach the number in the ‘E’ cell from the number in the ‘S’ cell in the visitation graph.

Name:___________________

The visit Priority is: UP, RIGHT, DOWN, LEFT



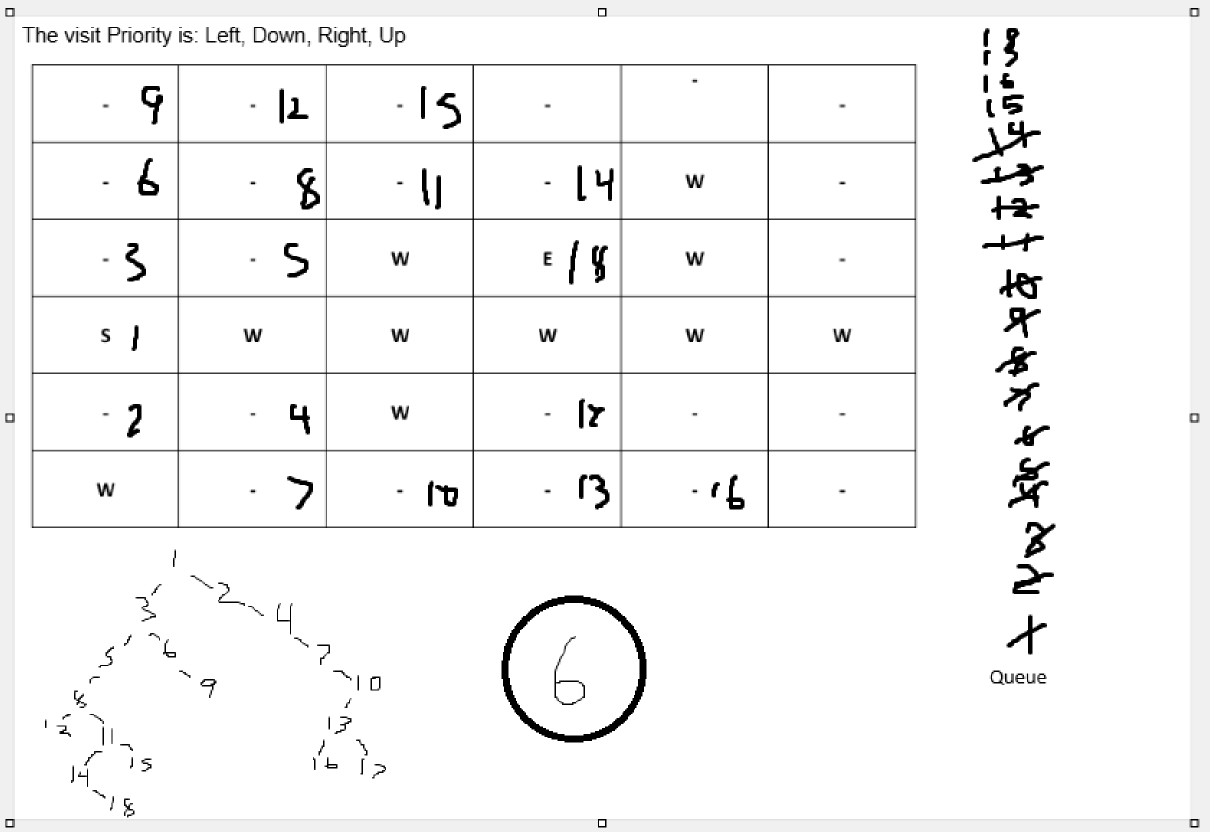
The visit Priority is: UP, RIGHT, DOWN, LEFT

- 7	- 10	- 13	- 16	-	-
- 4	- 8	- 11	- 14	W	-
- 2	- 5	W	E 17	W	-
S 1	W	W	W	W	W
- 3	- 6	W	-	-	-
W	- 9	- 12	- 15	-	-

17
16
15
~~14~~
~~13~~
~~12~~
~~11~~
~~10~~
~~9~~
~~8~~
~~7~~
~~6~~
~~5~~
4
~~3~~
~~2~~
~~1~~
Queue:

6

The visit Priority is: Left, Down, Right, Up



The visit Priority is: Left, Down, Right, Up

- 9	- 12	- 15	-	-	-
- 6	- 8	- 11	- 14	W	-
- 3	- 5	W	E 18	W	-
S 1	W	W	W	W	W
- 2	- 4	W	- 17	-	-
W	- 7	- 10	- 13	- 16	-

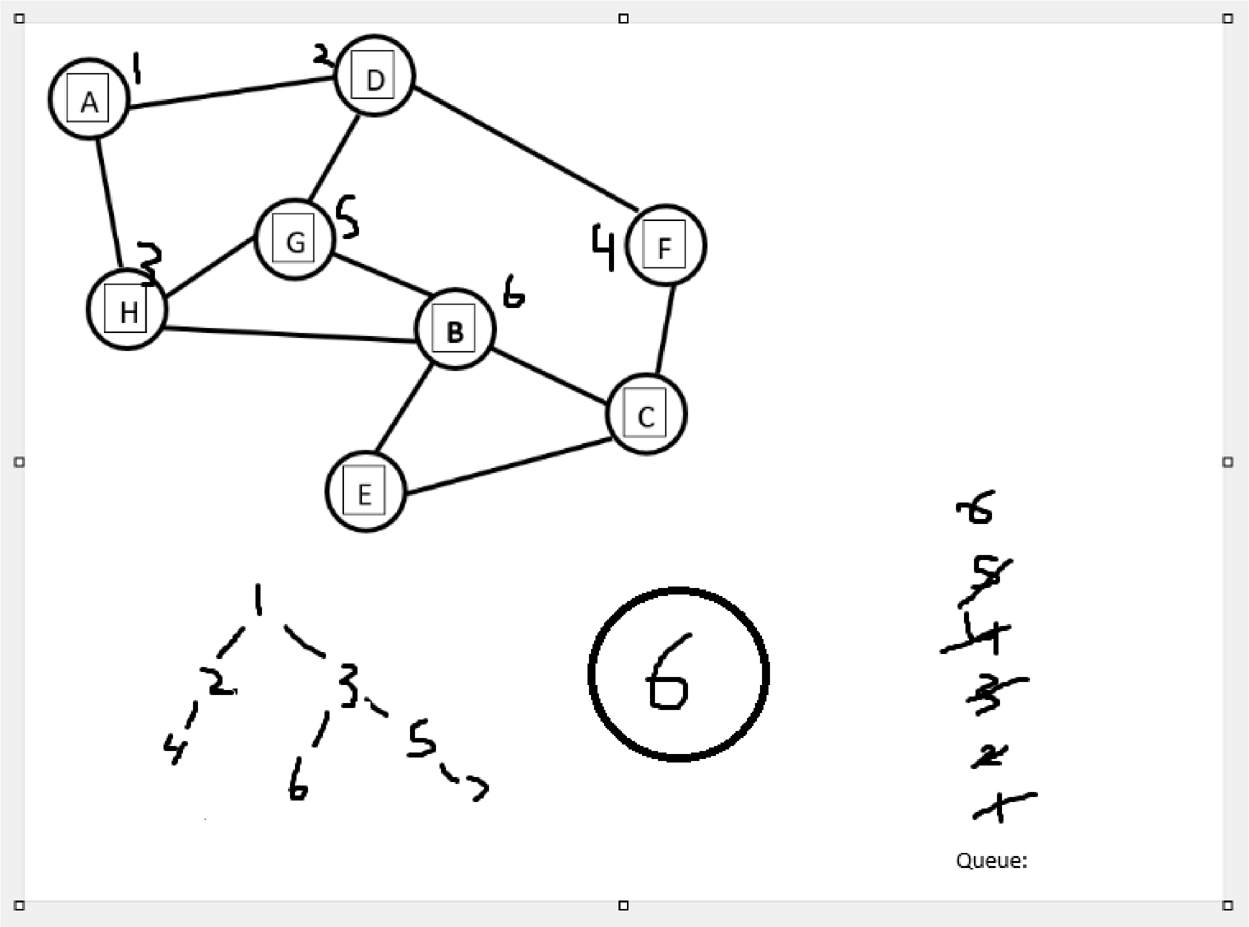
18
17
16
15
~~14~~
~~13~~
~~12~~
~~11~~
~~10~~
~~9~~
~~8~~
~~7~~
~~6~~
~~5~~
~~4~~
~~3~~
~~2~~
~~1~~
Queue

6

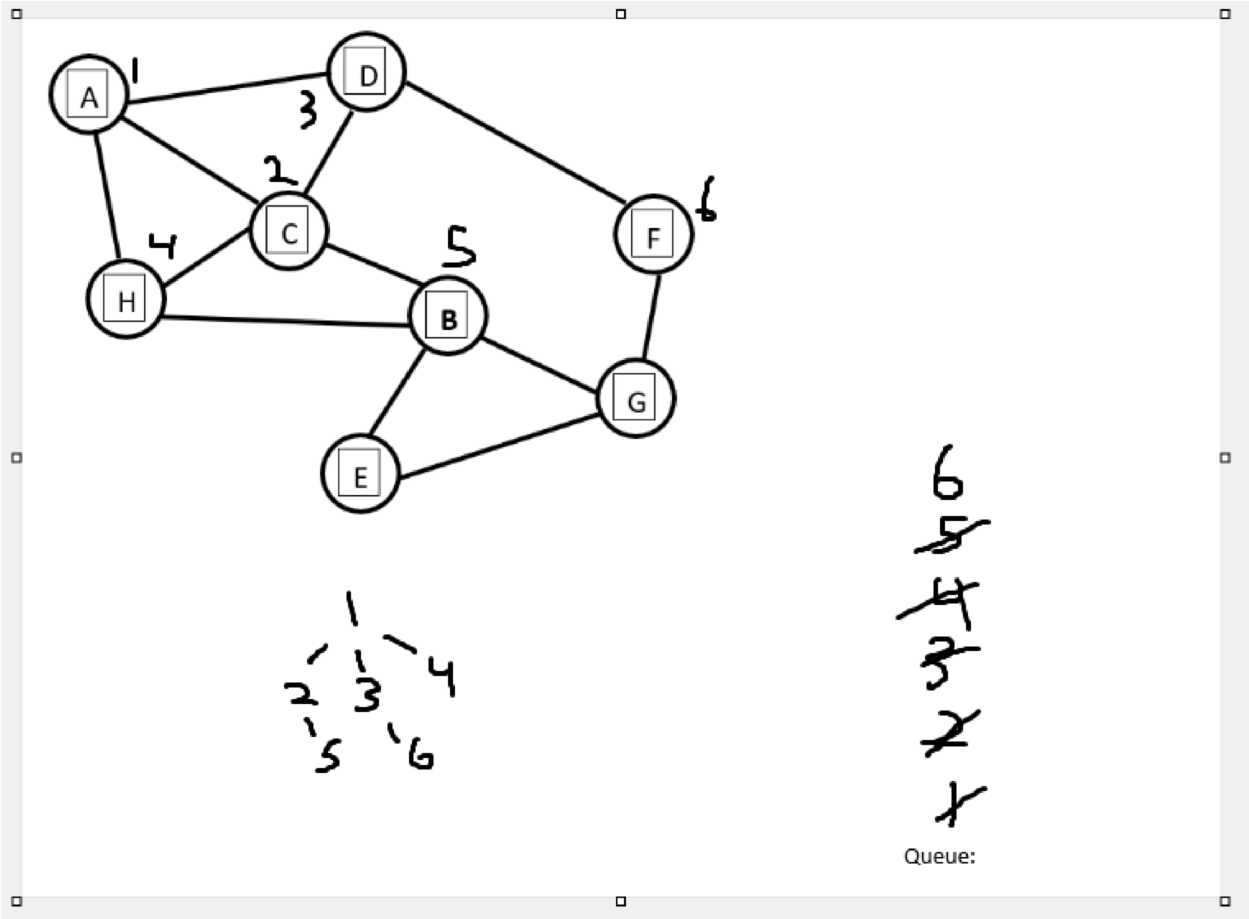
Name:___________________

Solve the following Maze using a BFS.

1. Create two areas below the Graph. One for showing your queue and the other for a visitation graph.
2. Write a 1 near the graph node with the label ‘A’, offer 1 to the queue and add a node with the text 1 to the visitation graph.
3. If the queue is empty go to step 8, otherwise go to step 4
4. Poll a number from the queue (referred to as **removed**)
5. If **removed** is the number next to the ‘C’ node go to step 8, otherwise go to step 6
6. For each node that is not numbered and is adjacent **popped**. (**Access nodes in alphabetical order)**
    a. Write a number next to the node using the next available ascending number, referred to as **neighbor**.
    b. Offer **neighbor** to the queue
    c. Create a node in the visitation graph with **neighbor**
    d. Connect the **neighbor** in the visitation graph to **removed**
7. Go to step 3
8. Write down the number of edges it takes to reach the number next to the ‘B’ node from the number next to the ‘A’ node in the visitation graph.



Name:____________________


A
1
D
3
2
C
H
4
B
5
F
6
G
E
1
2
3
4
5
6
6
~~5~~
~~4~~
~~3~~
~~2~~
~~1~~
Queue: