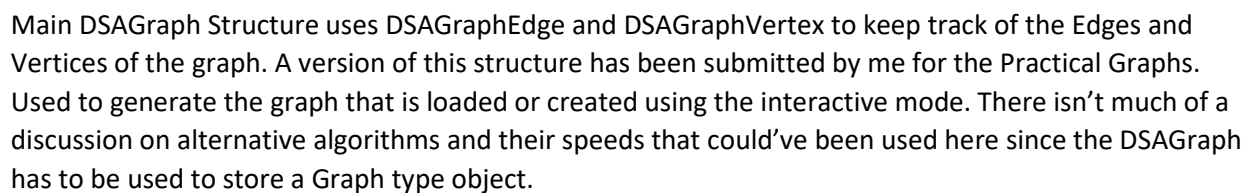
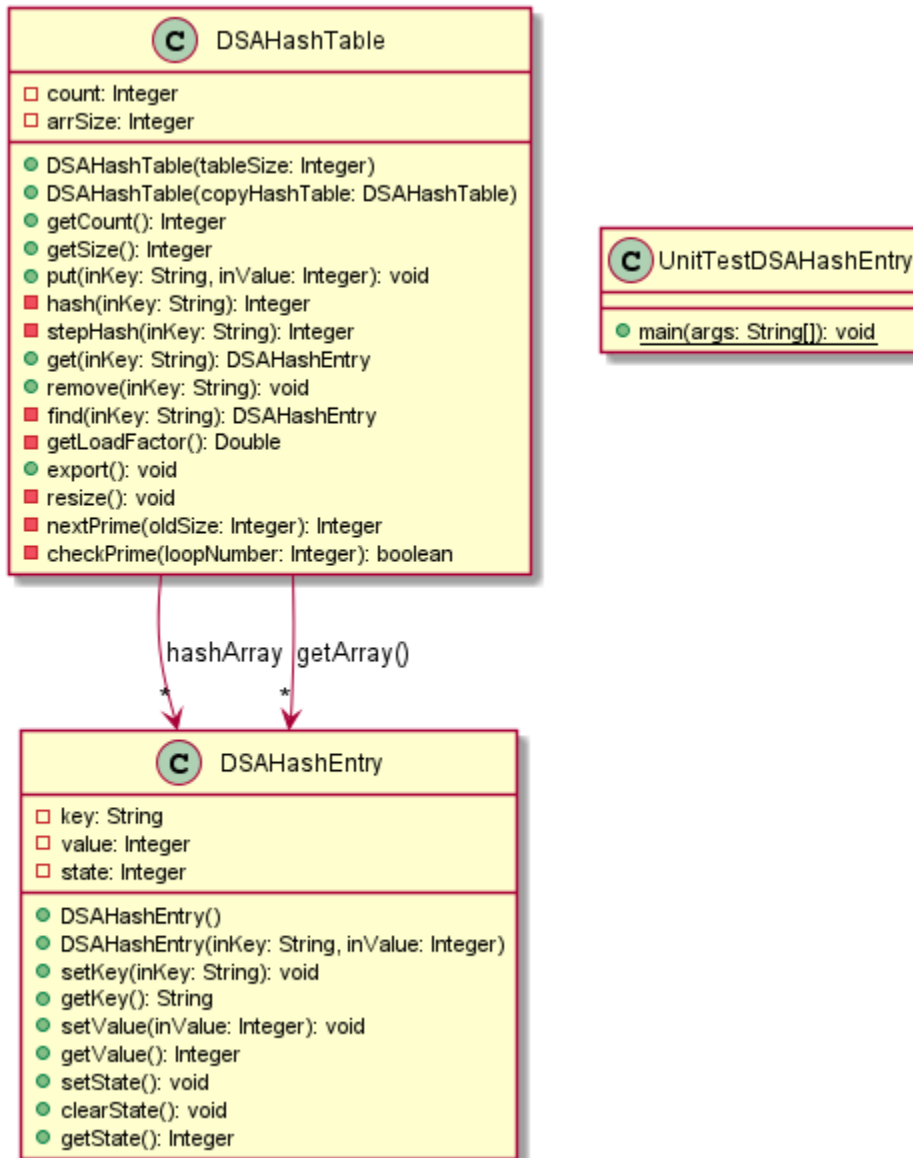


1) DSAGraph.java, DSAGraphEdge.java, DSAGraphVertex.java



- i) Uses the Sort.class to do a Bubble Sort on the Ranked Paths after they are generated.
- ii) Uses the Path.class to create the Paths and their cost to travel which will be stored in a Path[] array. The alternative could've been a 2D array, same effect, same speeds but this is more of an ordered way to store the structures.
- iii) Uses the LinkedList to keep track of the Vertices and Edges (this is according to the slides), no alternative could've been used.
- iv) LinkedList class is made up of LinkedListNodes (this is according to the slides)
- v) Iterable is implemented for the LinkedList so we can iterate through the LinkedLists.

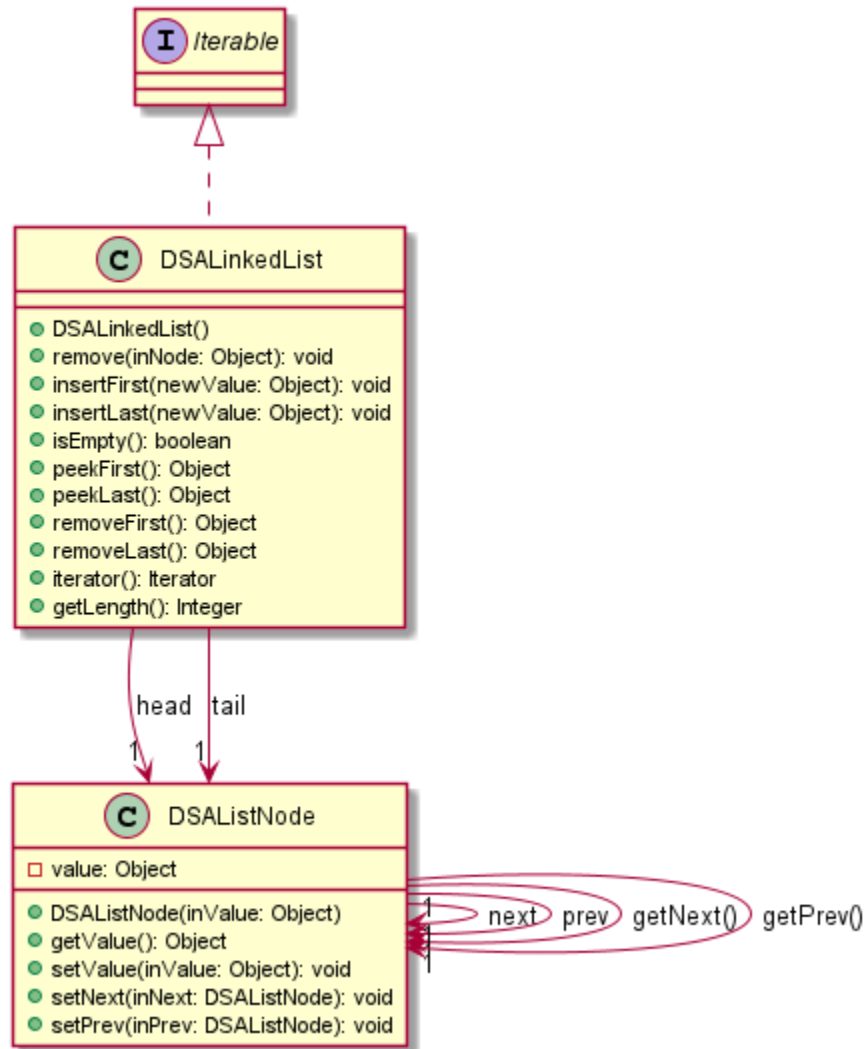
2) DSAHashTable.java, DSAHashEntry.java



Main DSAHashTable and DSAHashEntry class used to store the EdgeCodes and VerticeCodes, since the EdgeCodes and VerticeCodes need to be unique, I decided to use the EdgeCodes on a separate HashTable and VerticeCodes on a separate HashTable. The Code Character was used as a Key and the weight associated with it was used as a Value, creating the HashTable. Has been submitted before for the HashTable Practicals.

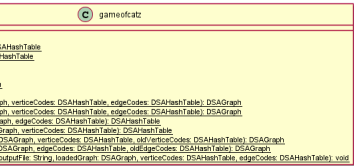
Could've used a LinkedList for the same effect, but the fetch speeds of the DSAHashTable are often $O(1)$ as opposed to the LinkedList where it could be $O(N)$ on average.

3) DSALinkedList.java, DSAListNode.java



DSALinkedList, DSAListNode is pretty standard implementation of the DSALinkedList referred to in the Lecture Slides. This is used for the Vertices and Edges in the DSAGraph because that is how the algorithm says we must do it in the Lecture Slides. This code has been previously submitted to the LinkedList Practicals. We could possibly use HashTables, for faster access speeds $O(1)$ as opposed to LinkedLists $O(N)$, but again as mentioned before, we have explicitly used DSALinkedLists in the Lecture Slides.

4) Gameofcatz.class



This is the class that will be running all of the previous classes in unison to generate paths for cats in simulation mode and would allow you to edit the cats world in Interactive Mode.

To run the code in Simulation Mode after compiling run

“java gameofcatz -s infile.txt savefile.txt” where infile.txt is the infile for the saved network file and the savefile.txt is where you want the ranked paths in order to be saved.

When running the code in Interactive Mode “java gameofcatz -i” you will see this

```
sahas@MSI:/mnt/c/Users/sahas/Curtin - Sem2/DSA - COMP/Assignment 1/Main App$ java gameofcatz -i
-----Going to interactive mode!-----
  ^ ^
( 0..0 ) meow!

-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice:
```

1. Load input file -> Will initialize a graph based on the input file that is given, much like what we do in simulation mode.
2. Node Operations -> Will show you a SubMenu as such :=

```
-----Node Submenu-----  
1. Delete Node  
2. Update Node  
3. Insert Node  
4. Display All Nodes  
5. Display All Node Codes  
6. Find Node  
7. Exit Node Sub Menu  
What would you like to do?: |
```

Here you will be allowed to Delete a Node, update a Node, Insert a new Node, Display all Nodes and their weights, Display all the Node Codes and Exit.

When it comes to Inserting a New Node or Updating a Node, you're not allowed to add any weight you desire, but you must choose one from the available registered Node Codes.

Finding a Node returns the Node Label and its weight.

3. Edge Operations -> Will show you a SubMenu as such :=

```
-----Edge Submenu-----  
1. Delete Edge  
2. Update Edge  
3. Insert Edge  
4. Display All Edges  
5. Display All Edge Codes  
6. Find Edge  
7. Exit Edge Sub Menu  
What would you like to do?: |
```

Here you will be allowed to Delete an Edge, Update an Edge, insert a new Edge, Display all Edges and their weight, Display all the Edge Codes and Exit.

When it comes to Inserting a New Edge or Updating an Edge the same rule applies as Inserting a new Node or Updating a Node, it must have to be registered as an Edge Code first.

4. Speaking of registering an Edge Code or a Node Code

```
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 5
Exiting Edge Code Tweaking menu!
Edge Parameters have changed, regenerating the graph.
```

```
-----Vertice Code Tweaking Submenu-----
1. Add Vertice Code Parameter
2. Delete Vertice Code Parameter
3. Update Vertice Code Parameter
4. Display Vertice Codes
5. Exit
```

If you want to manipulate the Edge and Vertice Codes this is the location to do it, we don't allow you to delete Edge Codes or Node Codes that are used by the Edges and Vertices as then there would be no backreference for them. Once you exit these two menus we regenerate the graph as if you turn the Edge Code "-" from weight 1 to 10, it impacts all the Edges that was based on Edge Code "-".

5. Display Graph option will display Weighted Matrix, here's an image how it is generated on the terminal.

```

-----Displaying Weighted Matrix-----
| |G|I|B|H|A|C|D|E|F|J|
|G| |1| | | |1|1| |1| |
|I|1| | |1| | | |1|1|
|B| | | |1|1| |1|1| |
|H| |1| | | | |1|1|1|
|A| | |1| | | |1| | |
|C|1| |1| | |1| |1| |
|D|1| | | |1| | | |
|E| | |1|1|1| | |1| |
|F|1|1|1|1| |1| |1| |
|J| |1| |1| | | | | |
File Name to Save?:

```

You get the question for a file name to save it immediately afterwards.

6. Display World will display the Adjacency List and will tell you the number of unique weights for the Vertices and Edges. Also, you can save this Adjacency List afterwards.

```

-----Displaying Adjacency!-----
| G | C D F I
| I | F G H J
| B | A E F C
| H | E F I J
| A | B E
| C | B F G D
| D | C G
| E | A B F H
| F | B C E G H I
| J | I H
Number of Unique Vertices: 4
Number of Unique Edges: 1
File Name to Save?: |

```

7. Display Routes option will immediately ask you for the Start and Destination, Generate and Display the routes in order on the Terminal and ask you where you would like to save these paths.

```

Your choice: 8
Starting position: A
Destination: B
AB : 1
AEB : 2
AEFB : 3
AEFCB : 4
AEFGCB : 4
AEFIGCB : 4
AEFGDCB : 5
AEFIGDCB : 5
AEHFB : 104
AEHIFB : 104
AEHIGCB : 104
AEHIGFB : 104
AEFHIGCB : 105
AEHFCB : 105
AEHFGCB : 105
AEHFIGCB : 105
AEHIFCB : 105
AEHIFGCB : 105
AEHIGCFB : 105
AEHIGDCB : 105
AEHIGFCB : 105
AEHJIFB : 105
AEHJIGCB : 105
AEHJIGFB : 105
AEFHIGDCB : 106
AEFHJIGCB : 106
AEHFGDCB : 106
AEHFIGDCB : 106
AEHIFGDCB : 106
AEHIGDCFB : 106
AEHJIFCB : 106
AEHJIFGCB : 106
AEHJIGCFB : 106
AEHJIGDCB : 106
AEHJIGFCB : 106
AEFHJIGDCB : 107
AEHJIFGDCB : 107
AEHJIGDCFB : 107
File Name to Save?:

```

As you can see, it is very neatly ordered and displays the “cost” of the path in the diagram. For the sample input provided we can clearly see the paths that are less than double digits are the paths where you don’t have to go through the dog (Node H who costs 100 to pass).

When calculating this value, we do not calculate the value for existing (although some philosophers will argue otherwise). That is if the Cat goes from A to B. We don’t calculate the cost of the cat going being at A and ending up at B. But only the cost of the destinations in between. Therefore, it only goes through the edge AB, so the cost of that path is one. This may be different in how you calculate it but regardless the result is the same as this logic is uniformly applied to all the paths.

8. Save Network, will just reverse generate a blueprint of the network (much like the sample input file we have generated), which can be used to initialize graphs from the program on a later date. Works in quite similar fashion to a “Save” function on a video game. You need to also type in a Start and Destination so you can run this file in simulation mode on a later occasion.

Here is a screenshot of what this generated file will look like ->

This is also why we make the assumption of Node Codes and Edge Codes having unique weights since after we generate a graph, inside a Node or a Edge we can only see the Label and Weight of it, but not the NodeCode or EdgeCode it might have utilized.

sahas - Notepad

File Edit Format View Help

```
# Node label code
# Edge label label code
# Ncode code weight
# Ecode code weight
# Start label
# Target label
NCode F -1
NCode T 1
NCode D 100
NCode - 0
Node G F
Node I F
Node B T
Node H D
Node A -
Node C -
Node D -
Node E -
Node F -
Node J -
ECode - 1
Edge A B -
Edge A E -
Edge B A -
Edge B E -
Edge B F -
Edge B C -
Edge C B -
Edge C F -
Edge C G -
Edge C D -
Edge D C -
Edge D G -
Edge E A -
Edge E B -
Edge E F -
Edge E H -
Edge F B -
Edge F C -
Edge F E -
Edge F G -
Edge F H -
Edge F I -
Edge G C -
Edge G D -
Edge G F -
Edge G I -
Edge H E -
Edge H F -
Edge H I -
Edge H J -
Edge I F -
Edge I G -
Edge I H -
Edge I J -
Edge J I -
Edge J H -
Start A
Target B
```

Traceability Matrix

		Requirements	Design/Code	Test
1	Modes	1.1 System displays usage if called without arguments	Gameofcatz.main()	UnitTestHarnesses/MainApp/README SIMULATION MODE 1
		1.2 System displays error message when you call simulation mode with a wrong	Gameofcatz.main()	UnitTestHarnesses/MainApp/README SIMULATION MODE 2
		3.2 System enters into Interactive mode	java gameofcatz -i	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 1
2	FileIO	2.1 System displays error when running the simulation mode where the infile doesn't exist	Gameofcatz.initializedInput()	UnitTestHarnesses/MainApp/README SIMULATION MODE 3
		2.2 System displays an error message in simulation mode when the infile is an empty file	Gameofcatz.initializedInput()	UnitTestHarnesses/MainApp/README SIMULATION MODE 4
		2.3 System displays an error message in simulation mode when the infile has a non-defined target and start	Gameofcatz.initializedInput()	UnitTestHarnesses/MainApp/README SIMULATION MODE 5 UnitTestHarnesses/MainApp/README SIMULATION MODE 6
		2.4 System displays an error but generates paths when the infile has Nodes and Edges that are defined by Node Code and Edge Codes that do not exist	Gameofcatz.initializedInput()	UnitTestHarnesses/MainApp/README SIMULATION MODE 8
		2.5 System ignores duplicate Node Codes and Edge Codes, generates file with paths	Gameofcatz.initializedInput()	UnitTestHarnesses/MainApp/README SIMULATION MODE 9
3	Gameplay	3.1 System displays a "no paths found" message in simulation mode when there are no paths	DSAGraph.depthFirstSearch()	UnitTestHarnesses/MainApp/README SIMULATION MODE 7
		3.2 Users inputs an invalid choice in the menus and it displays an error message	Gameofcatz.menu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 2
		3.3 Entering an Edge or a Node without having any Node or Edge Codes	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 3

		initialized displays an error message		
		3.4 Error message displayed when trying to delete an Edge or a Node when there are none in the graph	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 4
		3.5 Error message displayed when trying to update an Edge or a Node, when there are none.	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 4
		3.6 Error message displayed when trying to update a Node or an Edge when there are none in the graph.	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 5
		3.7 Adding an Edge Code or a Node Code but entering an illegal value	Gameofcatz.nodeCodeSubMenu() Gameofcatz.edgeCodeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 6
		3.8 Adding an Edge Code Successfully and Displaying it	Gameofcatz.nodeCodeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 7
		3.9 Adding a Node/Vertice Code Successfully and Displaying it	Gameofcatz.nodeCodeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 8
		3.10 Trying to add a Duplicate Node or an Edge	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 9
		3.11 Trying to add a Node or an Edge with a Node Code or an Edge Code that doesn't exist	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 10
		3.12 Trying to enter empty lines in the Menu (String Null Pointer error handling)	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu() Gameofcatz.nodeCodeSubMenu() Gameofcatz.edgeCodeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 11
		3.12 Loading an infile successfully using Menu Option 1	Gameofcatz.initializedInput()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 12
		3.13 Displaying the graph	DSAGraph.displayMatrix()	UnitTestHarnesses/DSAGraph/UnitTestDSAGraph line 55
		3.14 Printing the graph	DSAGraph.printDisplayMatrix()	UnitTestHarnesses/DSAGraph/UnitTestDSAGraph line 56
		3.15 Displaying the adjacency	DSAGraph.displayAdjacency()	UnitTestHarnesses/DSAGraph/UnitTestDSAGraph line 54

		3.16 Printing the adjacency	DSAGraph.printDisplayAdjacency()	UnitTestHarnesses/DSAGraph/UnitTestDSAGraph line 57
		3.17 Displaying the paths	DSAGraph.displayPaths()	UnitTestHarnesses/DSAGraph/UnitTestDSAGraph line 60
		3.18 Printing the paths	DSAGraph.printPaths()	UnitTestHarnesses/DSAGraph/UnitTestDSAGraph line 61
		3.19 Adding a Node or an Edge Successfully	Gameofcatz.nodeSubMenu() Gameofcatz.edgeSubMenu()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 13
		3.20 Printing out the Simulation	Gameofcatz.saveNetwork()	UnitTestHarnesses/MainApp/README INTERACTIVE MODE 14

There are three scenarios to be run ->

- 1) Loading a file using option 1, saving the Graph, saving the world, saving the routes, saving the network.
- 2) Loading a file using option 1, manipulating the graph, saving the Graph, saving the world, saving the routes, saving the network.
- 3) Manipulating a graph directly, saving the Graph, saving the world, saving the routes, saving the network.

1) Simulation 1

```
-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 1
Please enter an input file: infile.txt
Data loaded from infile.txt
-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: |
```

Enter number 1 as option.

Output Graph

```
-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 6
-----Displaying Weighted Matrix-----
| |A|B|C|D|E|F|G|H|I|J|
|A| |1| | |1| | | | |
|B|1| |1| |1|1| | | |
|C| |1| |1| |1|1| | |
|D| | |1| | | |1| | |
|E|1|1| | | |1| |1| |
|F| |1|1| |1| |1|1|1| |
|G| | |1|1| |1| | |1| |
|H| | | | |1|1| | |1|1|
|I| | | | | |1|1|1| |1|
|J| | | | | | |1|1| |
File Name to Save?: |
```

Enter name: simulation1graph.txt

```
-----Displaying Weighted Matrix-----
| |A|B|C|D|E|F|G|H|I|J|
|A| |1| | |1| | | | |
|B|1| |1| |1|1| | | |
|C| |1| |1| |1|1| | |
|D| | |1| | | |1| | |
|E|1|1| | | |1| |1| |
|F| |1|1| |1| |1|1|1| |
|G| | |1|1| |1| | |1| |
|H| | | | |1|1| | |1|1|
|I| | | | | |1|1|1| |1|
|J| | | | | | |1|1| |
File Name to Save?: simulation1graph.txt
```

File has been successfully generated (will leave inside the UnitTestHarnesses/MainApp directory).

simulation1graph - Notepad

	A	B	C	D	E	F	G	H	I	J
A		1			1					
B	1		1		1	1				
C		1		1		1	1			
D			1				1			
E	1	1				1		1		
F		1	1		1		1	1	1	
G			1	1		1			1	
H					1	1			1	1
I						1	1	1		1
J								1	1	

Choose display world option.

```
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 7
-----Displaying Adjacency!-----
| A | B E
| B | A E F C
| C | B F G D
| D | C G
| E | A B F H
| F | B C E G H I
| G | C D F I
| H | E F I J
| I | F G H J
| J | I H
Number of Unique Vertices: 4
Number of Unique Edges: 1
File Name to Save?:
```

Save file name as simulation1world.txt

simulation1world - Notepad


File Edit Format View Help

A	B E
B	A E F C
C	B F G D
D	C G
E	A B F H
F	B C E G H I
G	C D F I
H	E F I J
I	F G H J
J	I H

Choose display routes option, choose positions A and B

```
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 8
Starting position: A
Destination: B
AB : 1
AEB : 2
AEFB : 3
AEFCB : 4
AEFGCB : 4
AEFIGCB : 4
AEFGDCB : 5
AEFIGDCB : 5
AEHFB : 104
AEHIFB : 104
AEHIGCB : 104
AEHIGFB : 104
AEFHIGCB : 105
AEHFCB : 105
AEHFGCB : 105
AEHFIGCB : 105
AEHIFCB : 105
AEHIFGCB : 105
AEHIGCFB : 105
AEHIGDCB : 105
AEHIGFCB : 105
AEHJIFB : 105
AEHJIGCB : 105
AEHJIGFB : 105
AEFHIGDCB : 106
AEFHJIGCB : 106
AEHFGDCB : 106
AEHFIGDCB : 106
AEHIFGDCB : 106
AEHIGDCFB : 106
AEHJIFCB : 106
AEHJIFGCB : 106
AEHJIGCFB : 106
AEHJIGDCB : 106
AEHJIGFCB : 106
AEFHJIGDCB : 107
AEHJIFGDCB : 107
AEHJIGDCFB : 107
File Name to Save?:
```


Save ranked paths as simulation1routes.txt

 simulation1routes - Notepad

File Edit Format View Help

```
AB : 1
AEB : 2
AEFB : 3
AEFCB : 4
AEFGCB : 4
AEFIGCB : 4
AEFGDCB : 5
AEFIGDCB : 5
AEHFB : 104
AEHIFB : 104
AEHIGCB : 104
AEHIGFB : 104
AEFHIGCB : 105
AEHFCB : 105
AEHFGCB : 105
AEHFIGCB : 105
AEHIFCB : 105
AEHIFGCB : 105
AEHIGCFB : 105
AEHIGDCB : 105
AEHIGFCB : 105
AEHJIFB : 105
AEHJIGCB : 105
AEHJIGFB : 105
AEFHIGDCB : 106
AEFHJIGCB : 106
AEHFGDCB : 106
AEHFIGDCB : 106
AEHIFGDCB : 106
AEHIGDCFB : 106
AEHJIFCB : 106
AEHJIFGCB : 106
AEHJIGCFB : 106
AEHJIGDCB : 106
AEHJIGFCB : 106
AEFHJIGDCB : 107
AEHJIFGDCB : 107
AEHJIGDCFB : 107
```

Choose Save Network Option with Starting and Ending from A and B

```
-----Interactive Mode! Main Menu!-----  
Please enter the number for whatever respective option you want  
(1) Load input file  
(2) Node Operations (find, insert, delete, update)  
(3) Edge Operations (find, add, remove, update)  
(4) Edge Code tweaks  
(5) Node Code tweaks  
(6) Display graph  
(7) Display world  
(8) Display routes  
(9) Save Network  
(10) Exit  
Your choice: 9  
Starting position: A  
Destination: B  
File Name to Save?: simulation1network.txt
```

simulation1network - Notepad

```
File Edit Format View Help
# Node label code
# Edge label label code
# Ncode code weight
# Ecode code weight
# Start label
# Target label
Ncode F -1
Ncode T 1
Ncode - 0
Ncode D 100
Node A -
Node B T
Node C -
Node D -
Node E -
Node F -
Node G F
Node H D
Node I F
Node J -
Ecode - 1
Edge A B -
Edge A E -
Edge B A -
Edge B E -
Edge B F -
Edge B C -
Edge C B -
Edge C F -
Edge C G -
Edge C D -
Edge D C -
Edge D G -
Edge E A -
Edge E B -
Edge E F -
Edge E H -
Edge F B -
Edge F C -
Edge F E -
Edge F G -
Edge F H -
```

Exit!, End of Simulation1.

2) Simulation 2

Load infile.txt

```
-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 1
Please enter an input file: infile.txt
Data loaded from infile.txt
```

We are going to assume this cat can teleport, so all Edges now should have a weight of 0.

Go to Edge Code tweaks, Update Edge Code Parameter, choose the char “-” and enter its weight as 0.
Display Edge Codes to confirm tweak.

```
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 3
Please enter the Char for the Edge Code to be update: -
Please enter a new integer weight for the Edge Code: 0
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 4
EdgeCode:Weight
-:0
```

Now we will add another Edge Code for a Zone where the cat doesn't have its teleportation powers. Add Edge Code Parameter, Give this code name X, and a weight of 10.

```
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 1
Please enter the Char for the Edge Code: X
Please enter the weight for the Edge Code: 10
```

Display Edge Codes to confirm this change is recorded

```
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 4
EdgeCode:Weight
-:0
X:10
```

Exit, and go to Node Code Tweaks, Add a Vertice Code Parameter, R for a TRex and a weight of 200.
Display Vertice Codes to confirm this.

```
-----Vertice Code Tweaking Submenu-----
1. Add Vertice Code Parameter
2. Delete Vertice Code Parameter
3. Update Vertice Code Parameter
4. Display Vertice Codes
5. Exit
What is your choice?: 1
Please enter the Char for the Edge Code: R
Please enter the weight for the Edge Code: 200
-----Vertice Code Tweaking Submenu-----
1. Add Vertice Code Parameter
2. Delete Vertice Code Parameter
3. Update Vertice Code Parameter
4. Display Vertice Codes
5. Exit
What is your choice?: 4
EdgeCode:Weight
F:-1
T:1
-:0
D:100
R:200
```

Exit. And go to Node Operations. Add a Node Z with a weight of R (For the TRex).

```
-----Node Submenu-----
1. Delete Node
2. Update Node
3. Insert Node
4. Display All Nodes
5. Display All Node Codes
6. Find Node
7. Exit Node Sub Menu
What would you like to do?: 3
New name for this node?: Z
NodeCodes:Weight
F:-1
T:1
-:0
D:100
R:200
New weight code for this node?: R
-----Node Submenu-----
```

Now Display All Nodes.

```
-----Node Submenu-----
1. Delete Node
2. Update Node
3. Insert Node
4. Display All Nodes
5. Display All Node Codes
6. Find Node
7. Exit Node Sub Menu
What would you like to do?: 4
-----Displaying all Nodes!-----
Node Label : Node Value
G : -1
I : -1
B : 1
A : 0
C : 0
D : 0
E : 0
F : 0
J : 0
H : 100
Z : 200
```

Exit Node Sub Menu. Go to Edge Operations since we have no way of visiting this TRex.

```

-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 3

```

Insert Edge H Z with weight code X and and J Z with weight code X (cat can't teleport when visiting the TRex) and Display All the Edges

```

-----Edge Submenu-----
1. Delete Edge
2. Update Edge
3. Insert Edge
4. Display All Edges
5. Display All Edge Codes
6. Find Edge
7. Exit Edge Sub Menu
What would you like to do?: 3
What is the 'from' of the edge you want to add?: H
What is the 'to' of the edge you want to add?: Z
EdgeCodes:Weight
-:0
X:10
New weight code for this Edge?: X
-----Edge Submenu-----
1. Delete Edge
2. Update Edge
3. Insert Edge
4. Display All Edges
5. Display All Edge Codes
6. Find Edge
7. Exit Edge Sub Menu
What would you like to do?: 3
What is the 'from' of the edge you want to add?: J
What is the 'to' of the edge you want to add?: Z
EdgeCodes:Weight
-:0
X:10
New weight code for this Edge?: X

```

```

What would you like to do?: 4
-----Displaying all Edges!-----
Edge From -> Edge To : Edge value
A -> B : 0
A -> E : 0
B -> A : 0
B -> E : 0
B -> F : 0
B -> C : 0
C -> B : 0
C -> F : 0
C -> G : 0
C -> D : 0
D -> C : 0
D -> G : 0
E -> A : 0
E -> B : 0
E -> F : 0
E -> H : 0
F -> B : 0
F -> C : 0
F -> E : 0
F -> G : 0
F -> H : 0
F -> I : 0
G -> C : 0
G -> D : 0
G -> F : 0
G -> I : 0
H -> E : 0
H -> F : 0
H -> I : 0
H -> J : 0
I -> F : 0
I -> G : 0
I -> H : 0
I -> J : 0
J -> I : 0
J -> H : 0
H -> Z : 10
J -> Z : 10
-----Edge Submenu-----

```

As you can see with the code – have a weight of 0 since we updated it.

We'll assume the dog has been eaten by the TRex, so we will delete the Dog Node (H), this should remove all related edges as well (including the one we created)

Go to Node Operations. Delete Node, enter H. and then display all Nodes to confirm it has been deleted.


```

-----Node Submenu-----
1. Delete Node
2. Update Node
3. Insert Node
4. Display All Nodes
5. Display All Node Codes
6. Find Node
7. Exit Node Sub Menu
What would you like to do?: 1
What would you like to delete?: H
-----Node Submenu-----
1. Delete Node
2. Update Node
3. Insert Node
4. Display All Nodes
5. Display All Node Codes
6. Find Node
7. Exit Node Sub Menu
What would you like to do?: 4
-----Displaying all Nodes!-----
Node Label : Node Value
G : -1
I : -1
B : 1
A : 0
C : 0
D : 0
E : 0
F : 0
J : 0
Z : 200

```

Lets Exit this SubMenu, Go to Edge Operations again, and Display all Edges.

```

-----Displaying all Edges!-----
Edge From -> Edge To : Edge value
A -> B : 0
A -> E : 0
B -> A : 0
B -> E : 0
B -> F : 0
B -> C : 0
C -> B : 0
C -> F : 0
C -> G : 0
C -> D : 0
D -> C : 0
D -> G : 0
E -> A : 0
E -> B : 0
E -> F : 0
F -> B : 0
F -> C : 0
F -> E : 0
F -> G : 0
F -> I : 0
G -> C : 0
G -> D : 0
G -> F : 0
G -> I : 0
I -> F : 0
I -> G : 0
I -> J : 0
J -> I : 0
J -> Z : 10

```

As you can see all the links to the Dog (Node H) along with it has gone extinct.

Now lets exit this submenu and Display the Graph and name it as simulation2graph.txt

```

-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 6
-----Displaying Weighted Matrix-----
| G|I|B|A|C|D|E|F|J|Z|
|G| 0|  |  | 0|0|  |  |  |
|I|0|  |  |  |  | 0|0|  |
|B|  |  | 0|0|  | 0|0|  |
|A|  | 0|  |  | 0|  |  |
|C|0| 0|  | 0| 0|  |  |
|D|0|  |  | 0|  |  |  |
|E|  | 0|0|  |  | 0|  |
|F|0|0|0| 0| 0|  |  |
|J| 0|  |  |  |  |  |10|
|Z|  |  |  |  |  |  |  |
File Name to Save?: simulation2graph.txt

```

Display the World and save it as simulation2world.txt

```

-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 7
-----Displaying Adjacency!-----
| G | C D F I
| I | F G J
| B | A E F C
| A | B E
| C | B F G D
| D | C G
| E | A B F
| F | B C E G I
| J | I Z
| Z |
Number of Unique Vertices: 5
Number of Unique Edges: 2
File Name to Save?: simulation2world.txt

```

Display all the paths to the dinosaur (Node Z) Choose Display Routes and then starting position A and Finishing Position Z, save to simulation2routes.txt


```
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 8
Starting position: A
Destination: Z
AEFCGIJZ : 8
AEFCDCIJZ : 8
AEFGIJZ : 8
ABEFCGIJZ : 9
ABEFCDCIJZ : 9
ABEFGIJZ : 9
ABFCGIJZ : 9
ABFCDCIJZ : 9
ABFGIJZ : 9
ABFCGIJZ : 9
ABCGFIJZ : 9
ABCGIJZ : 9
ABCDGFIJZ : 9
ABCDGIJZ : 9
AEBFCGIJZ : 9
AEBFCDCIJZ : 9
AEBFGIJZ : 9
AEBFCGIJZ : 9
AEBCGFIJZ : 9
AEBCGIJZ : 9
AEBDCGFIJZ : 9
AEBDCGIJZ : 9
AEFBCGIJZ : 9
AEFBCDCIJZ : 9
AEFIJZ : 9
ABEFIJZ : 10
ABFIJZ : 10
ABCFIJZ : 10
AEBFIJZ : 10
AEBCFIJZ : 10
File Name to Save?: simulation2routes.txt
```

simulation2routes - Notepad

File Edit Format View Help

```
AEFCGIJZ : 8
AEFCDCGIJZ : 8
AEFGIJZ : 8
ABEFCGIJZ : 9
ABEFCDCGIJZ : 9
ABEFGIJZ : 9
ABFCGIJZ : 9
ABFCDCGIJZ : 9
ABFGIJZ : 9
ABFCGIJZ : 9
ABCGFIJZ : 9
ABCGFIJZ : 9
ABCGIJZ : 9
ABCDGFIJZ : 9
ABCDGIJZ : 9
AEBFCGIJZ : 9
AEBFCDCGIJZ : 9
AEBFGIJZ : 9
AEBFCGIJZ : 9
AEBCGFIJZ : 9
AEBCGFIJZ : 9
AEBCGIJZ : 9
AEBDCGFIJZ : 9
AEBDCGIJZ : 9
AEFBCGIJZ : 9
AEFBCDCGIJZ : 9
AEFIJZ : 9
ABEFIJZ : 10
ABFIJZ : 10
ABCFIJZ : 10
AEBFIJZ : 10
AEBCFIJZ : 10
```

Save Network, Starting position A, Ending Position Z. Save to simulation2network.txt

 simulation2network - Notepad

```
File Edit Format View Help
# Node label code
# Edge label label code
# Ncode code weight
# Ecode code weight
# Start label
# Target label
Ncode F -1
Ncode T 1
Ncode - 0
Ncode D 100
Ncode R 200
Node G F
Node I F
Node B T
Node A -|
Node C -
Node D -
Node E -
Node F -
Node J -
Node Z R
Ecode - 0
Ecode X 10
Edge A B -
Edge A E -
Edge B A -
Edge B E -
Edge B F -
Edge B C -
Edge C B -
Edge C F -
Edge C G -
Edge C D -
Edge D C -
Edge D G -
Edge E A -
Edge E B -
Edge E F -
Edge F B -
Edge F C -
Edge F E -
Edge F G -
```

3) Simulation 3

We do not load an input file, we will be creating a Graph from scratch.

Go to edge code tweaks, add Edge Code Parameters, Edge code “-” with a weight of 1, edge code “X” with a weight of 2.

```
-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 4
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 1
Please enter the Char for the Edge Code: -
Please enter the weight for the Edge Code: 1
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 1
Please enter the Char for the Edge Code: X
Please enter the weight for the Edge Code: 2
```

Display to confirm they have been registered.

```
-----Edge Code Tweaking Submenu-----
1. Add Edge Code Parameter
2. Delete Edge Code Parameter
3. Update Edge Code Parameter
4. Display Edge Codes
5. Exit
What is your choice?: 4
EdgeCode:Weight
-:1
X:2
```

Exit the Edge Code Tweaking Submenu, and go to Node Code Tweaks, add Node Code Tweaks, Node code with a Node Code of "F" with a weight of -2, and Node Code of "B" with a weight of 5.

```
-----Vertice Code Tweaking Submenu-----
1. Add Vertice Code Parameter
2. Delete Vertice Code Parameter
3. Update Vertice Code Parameter
4. Display Vertice Codes
5. Exit
What is your choice?: 1
Please enter the Char for the Edge Code: F
Please enter the weight for the Edge Code: -2
-----Vertice Code Tweaking Submenu-----
1. Add Vertice Code Parameter
2. Delete Vertice Code Parameter
3. Update Vertice Code Parameter
4. Display Vertice Codes
5. Exit
What is your choice?: 1
Please enter the Char for the Edge Code: B
Please enter the weight for the Edge Code: 5
```

Display Vertice Codes to confirm.

```
-----Vertice Code Tweaking Submenu-----
1. Add Vertice Code Parameter
2. Delete Vertice Code Parameter
3. Update Vertice Code Parameter
4. Display Vertice Codes
5. Exit
What is your choice?: 4
EdgeCode:Weight
F:-2
B:5
```

Exit the Vertice Code Tweaking Submenu, and enter the Node Operations menu.


```

-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 2
-----Node Submenu-----
1. Delete Node
2. Update Node
3. Insert Node
4. Display All Nodes
5. Display All Node Codes
6. Find Node
7. Exit Node Sub Menu
What would you like to do?: |

```

Insert the Nodes A, with weight code F, B with weight code B, C with weight code F, D with weight code B. Display all Nodes to confirm that they have been registered.

```

-----Node Submenu-----
1. Delete Node
2. Update Node
3. Insert Node
4. Display All Nodes
5. Display All Node Codes
6. Find Node
7. Exit Node Sub Menu
What would you like to do?: 4
-----Displaying all Nodes!-----
Node Label : Node Value
A : -2
B : 5
C : -2
D : 5

```

Exit the Node sub Menu and enter the Edge Operations Menu.

Insert an Edge from A to B, A to C and A to D all with weight code of -. Insert an Edge from B to A, B to C and B to D with a weight of X. Insert an Edge from C to A, C to B and C to D with a weight of -. And then update C to A edge from - to X. And Delete the Edge A to D. And display.

```

-----Displaying all Edges!-----
Edge From -> Edge To : Edge value
A -> B : 1
A -> C : 1
B -> A : 2
B -> C : 2
B -> D : 2
C -> B : 1
C -> D : 1
C -> A : 2

```

As you can see, the edge from A to D has been deleted. C to A edge has an updated weight code of X (weight 2). And then all the edges have been successfully been created.

Exit Edge Submenu. And Display Graph. Save as simulation3graph.txt

```
-----Displaying Weighted Matrix-----
| |A|B|C|D|
|A| |1|1| |
|B|2| |2|2|
|C|2|1| |1|
|D| | | | |
File Name to Save?: simulation3graph.txt
```

simulation3graph - Notepad

File Edit Format View Help

```
| |A|B|C|D|
|A| |1|1| |
|B|2| |2|2|
|C|2|1| |1|
|D| | | | |
```


Display world and save as simulation3world.txt

```
-----Displaying Adjacency!-----
| A | B C
| B | A C D
| C | B D A
| D |
Number of Unique Vertices: 2
Number of Unique Edges: 2
File Name to Save?: simulation3world.txt
```

```
| |A|B|C|
|B|A|C|D|
|C|B|D|A|
|D|
```

Display routes, choose from A to D. And save as simulation3routes.txt

```
-----Interactive Mode! Main Menu!-----
Please enter the number for whatever respective option you want
(1) Load input file
(2) Node Operations (find, insert, delete, update)
(3) Edge Operations (find, add, remove, update)
(4) Edge Code tweaks
(5) Node Code tweaks
(6) Display graph
(7) Display world
(8) Display routes
(9) Save Network
(10) Exit
Your choice: 8
Starting position: A
Destination: D
ACD : -2
ABCD : 5
ACBD : 5
ABD : 6
File Name to Save?: simulation3routes.txt
```

 simulation3routes - Notepad

File Edit Format View Help

```
ACD : -2
ABCD : 5
ACBD : 5
ABD : 6
```

Save Network and set starting position as A and ending position as D as simulation3network.txt and Exit.

simulation3network - Notepad

```
File Edit Format View Help
# Node label code
# Edge label label code
# Ncode code weight
# Ecode code weight
# Start label
# Target label
Ncode F -2
Ncode B 5
Node A F
Node B B
Node C F
Node D B
Ecode - 1
Ecode X 2
Edge A B -
Edge A C -
Edge B A X
Edge B C X
Edge B D X
Edge C B -
Edge C D -
Edge C A X
Start A
Target D
```

Conclusion and Future Work

- 1) The speed of the program should be improved, perhaps by implementing a faster sorting algorithm than the Bubble Sort that is implemented in this program. The quick sort seems to be ideal, however I had already finished most of the coding for the assignment by the time the quick sort was taught to us. For the file gameofcatz2.txt on simulation it takes close to 10 minutes to find all the paths and rank them, which in my opinion is fair, since it is more than 100,000+ paths. Fixing the sort should be able to bring it down by quite a time. All the other input files (gameofcatz.txt, and the amazing.txt series) run within very reasonable time and all their outputs (dubbed as "out-{filename}.txt") are left inside the UnitTestHarnesses/MainApp folder for you to view.
- 2) The DSAGraph and the DepthFirstSearch code has been written according to Lecture Slide standards so I doubt there is an improvement that can be made there (perhaps there is, I haven't come across such), if anything we can substitute the DSALinkedList for some HashTables for the $O(1)$ speed on access.

- 3) I would've liked more time on this assignment to foolproof the code a lot more, I think I have done a lot of intuitive ways that users may try to crash the code, but I personally would've preferred more Exception Handling routines to be used.
- 4) Add a gameplay like function where you can control the cat with a 2D representation of the world.
- 5) If there are Node Codes and Edge Codes with the same weight (there have not been any in the input files we have received as of 12/10), when saving the network it cannot identify which Node Code or Edge Code is. To solve this we could have added another attribute in the Edge and Node class that can store the Node Code and Edge Code of type String. Makes things a lot faster, as we can use the HashTable to immediately get the value which would be speed $O(1)$. Right now, we go through a loop of each element and get the relevant Label out which is on average speeds of $O(N)$.
- 6) While I can insert a file name and load a graph from that file and manipulate it, I cannot create a graph from scratch, and then load a graph on top of it. When you do it, I have made it so that the custom-made graph is erased, and then we start with the graph that was loaded in from the file. This could be an additional feature to be implemented.

NOTE: I was not aware we had to be accommodating of Labels for Nodes that are more than a Char (eg: A is a valid Label, but AB is not a valid label) I only knew of this when the snakesandladders1.txt file was added. Therefore, I have made a slight change when printing paths. To print a -> to show the start and the end of the Nodes (mainly because Node 34 can be interpreted as Node 3 and Node 4).

An image is attached.

outsnesandladders1 - Notepad

File Edit Format View Help

```
1->2->3->4->5->16->17->18->19->20->31->32->33->34 : 11
1->12->13->14->15->16->17->18->19->20->31->32->33->34 : 11
1->12->13->14->15->23->24->6->7->4->5->16->17->18->19->20->31->32->33->34 : 13
1->12->13->14->15->23->24->25->26->27->28->29->30->31->32->33->34 : 14
1->12->13->14->15->23->24->6->7->8->9->10->2->3->4->5->16->17->18->19->20->31->32->33->34 : 18
1->2->3->4->5->6->7->8->9->10->11->12->13->14->15->16->17->18->19->20->31->32->33->34 : 22
1->2->3->4->5->6->7->8->9->10->11->22->23->24->25->26->27->28->29->30->31->32->33->34 : 22
1->2->3->4->5->16->17->18->19->20->21->22->23->24->25->26->27->28->29->30->31->32->33->34 : 22
1->2->3->4->5->16->17->18->19->20->21->13->14->15->23->24->25->26->27->28->29->30->31->32->33->34 : 22
1->12->13->14->15->16->17->18->19->20->21->22->23->24->25->26->27->28->29->30->31->32->33->34 : 22
1->2->3->4->5->6->7->8->9->10->11->12->13->14->15->23->24->25->26->27->28->29->30->31->32->33->34 : 25
1->2->3->4->5->6->7->8->9->10->11->12->13->14->15->16->17->18->19->20->21->22->23->24->25->26->27->28->29->30->31->32->33->34 : 33
```