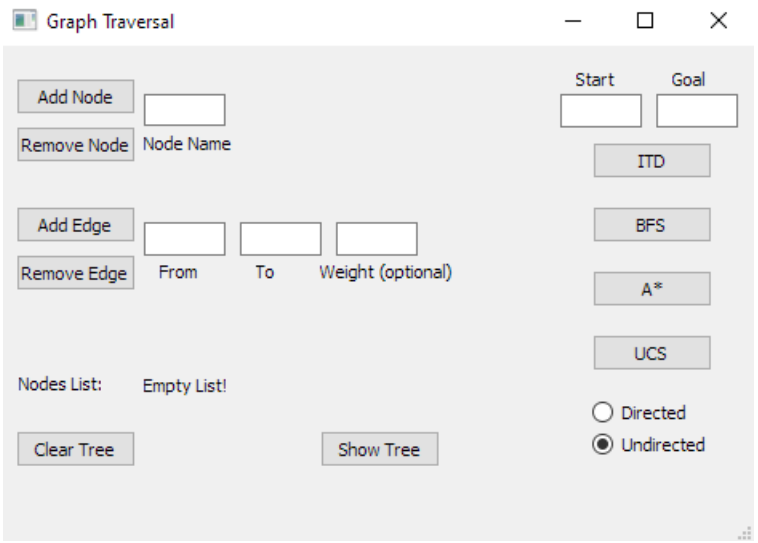
## User Guide:

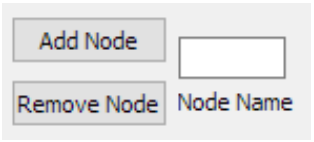
The application is named “Graph Traversal” 


After running the project, a user-friendly GUI screen appears as in the figure below:




Of course, each button has a specific function, and each text box should contain certain inputs. So, so we will explain each functionality in detail and with corresponding examples with figures.
Each graph is composed from nodes and edges.

This figure shows two buttons and one text box and a label for that text box.




1) The white text box of label “**Node Name**” is used to enter the name of the node being added.

2) The “**Add Node**” button is specified to add the node with name typed in the “**Node Name**” text box.

3) The “**Remove Node**” button is specified to remove the node with name typed in the “**Node Name**” text box.

**********

Each graph also “may” contain weighted edges.

This figure shows the options of dealing with edges in the program.

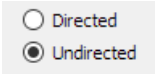

**Note:** you can easily create new nodes by creating edges instead of using the previous functionality to add nodes before creating edges.

1) The text box named “**From**” is used to enter the node name that the edge starts from.
2) The text box named “**To**” is used to enter the node name that the edge ends at.
3) The text box named “**Weight (optional)**” is used to enter the weight of the entered edges.
   **Note:** doesn’t differ which node to type in “**From**” and which to type in “**To**” if the graph is undirected, only differs if the graph is directed.
4) The “**Add Edge**” button is used to add the edge from the node entered in “**From**” to the node entered in “**To**” to the graph.
5) The “**Remove Edge**” button is used to delete the edge from the node typed in “**From**” to the node typed in “**To**” from the selected graph.

**********

Each graph can either be directed or undirected.

This figure contains two check buttons.




1) While operating on the directed graph, you must choose the “**Directed**” button.
2) While operating on the undirected graph, you must choose the “**Undirected**” button.

**********

Each graph can be drawn individually.

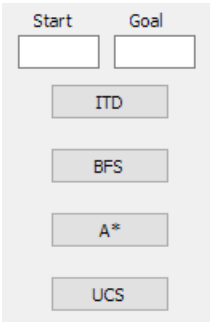
This figure shows the buttons that are responsible for drawing and clearing the graph.


1) So, any added node(s) will be added in a list that will appear next to the label "**Nodes List**" automatically by adding any node(s) using the "**Add Node**" or the "**Add Edges**" buttons, also nodes that are removed from the graph will be removed from the list.
   We initially have no nodes so the message "**Empty List!**" appears next to the "**Nodes list**" label.
2) After adding nodes and edges, if you want to see the graph you will click the "**Show Graph**" button.
3) After adding nodes and edges, if you want to delete the graph that you created, you will only have to click the "**Clear Graph**" button.

**********

We can apply search techniques on a graph. To apply the search techniques, we need to have a node to start from, and one or more goal nodes we need to reach.

This figure shows two text boxes of corresponding upper labels, and four buttons.



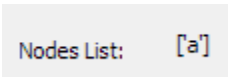

1) In the "**Start**" text box we must enter the starting node name, that we need to start the search from.
2) In the "**Goal**" text box we must type the goal(s) we want to reach. Note than in case of many nodes, we separate each node from the others by a single space.
3) To apply the Iterative-Deepening search technique and draw the final graph that shows the visited nodes path, we have to click the "**ITD**" button.
4) To apply the Breadth-First search technique and draw the final graph that shows the visited nodes path, we have to click the "**BFS**" button.
5) To apply the A star search technique and draw the final graph that shows the final path, we have to click the "**A***" button.

6) To apply the uniform cost search technique and draw the final graph that shows the final path, we have to click the "**UCS**" button.

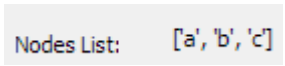
**********

Example of how the project visualize the graph:

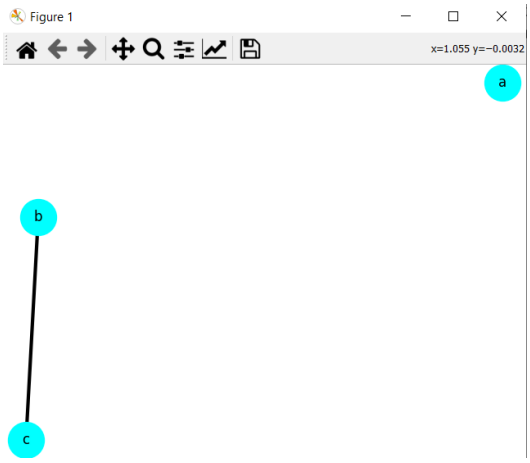
A node named 'a' is added.




An edge from node name 'b' to a node named 'c' is added.

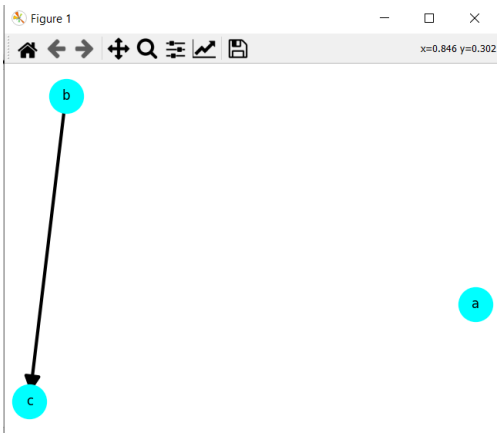



Visualization of the graph appears when clicking on "**Show Graph**" Button and "**Undirected**" radio button is checked.




**Note** that the nodes and the edges drawings visualization are randomly placed in the figure.

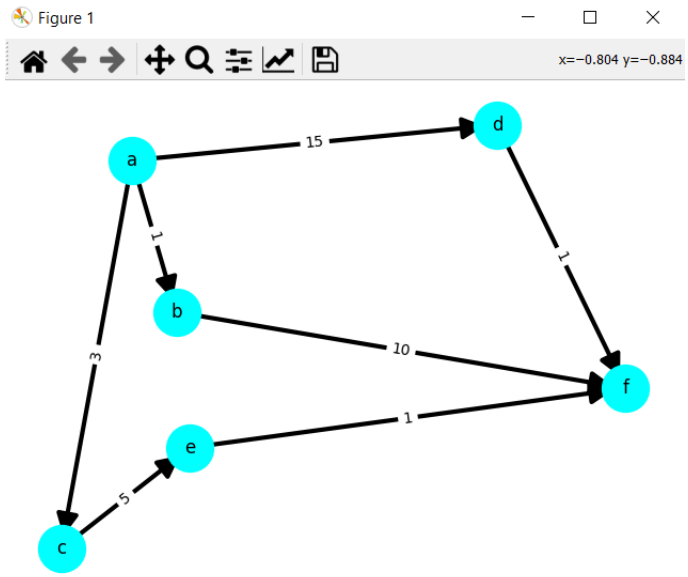
Visualization of the graph appears when clicking on "**Show Graph**" Button and the "**Directed**" radio button is checked.



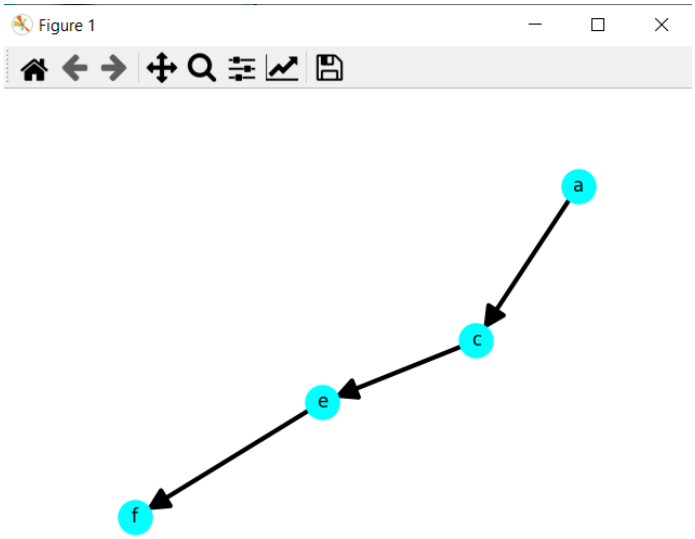

**Note** that the nodes and the edges drawings visualization are randomly placed in the figure.

## Example 1 using search techniques (UCS or A*) :

If we have the following graph:



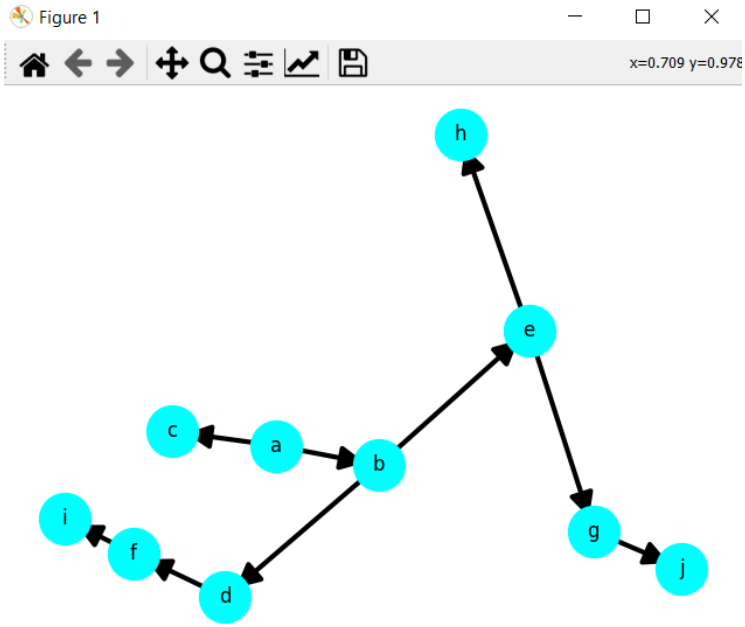

*Given this directed graph that has nodes **a**, **b**, **c**, **d**, **e** and **f***



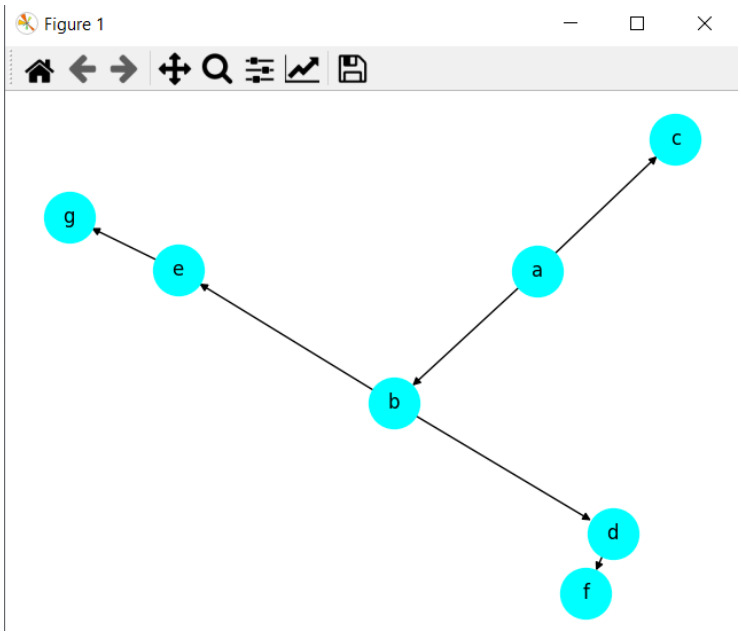

If we apply the **UCS** or the **A*** search technique, this figure appears with the final path drawn from **a** to **f** crossing **c** and **e** respectively.

## Example 2 using search techniques (ITD or BFS):

If we have the following graph:

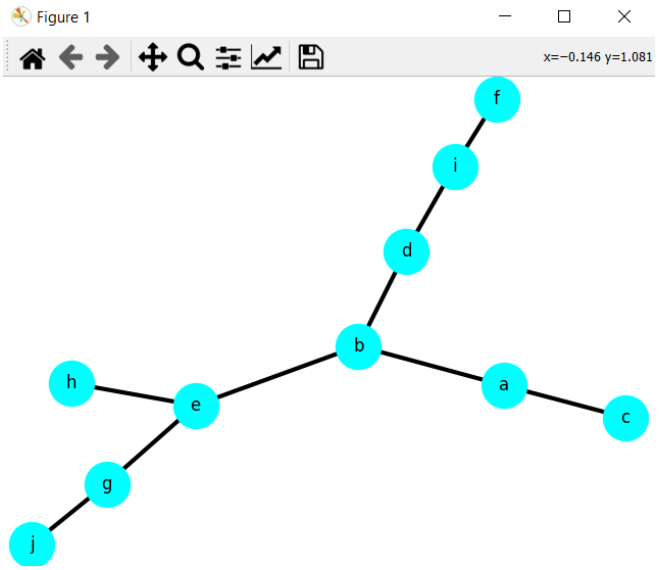



*Given this directed graph that has nodes **a**, **b**, **c**, **d**, **e, f, g, h, i and j***

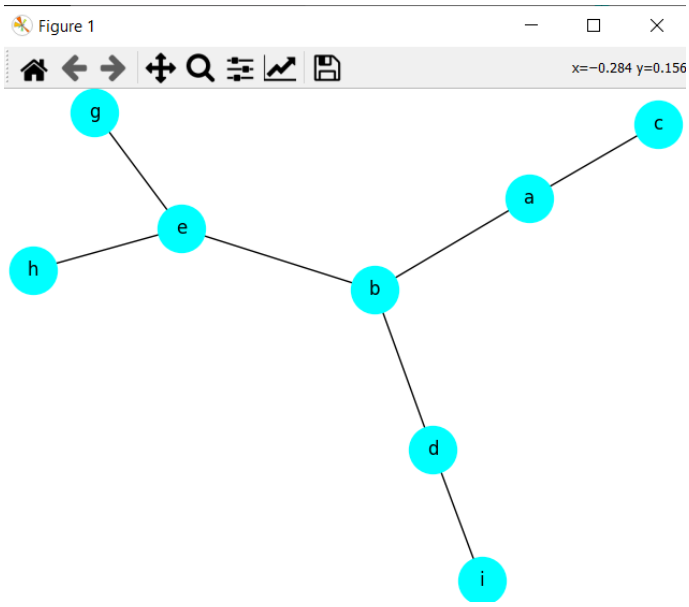



This figure appears when we apply the **BFS** technique from node **a** to node **g**. It represents the graph of the visited nodes while looking for the goal.

## Example 3 using search techniques (ITD or BFS) on an undirected graph :




*Given this graph that contains the nodes **a** to **j***




If we apply the **BFS** or the **ITD** search techniques, we will get this graph that represents the entire taken path from node **b** to node **g.**