A Manual to Efficient Space Traversal

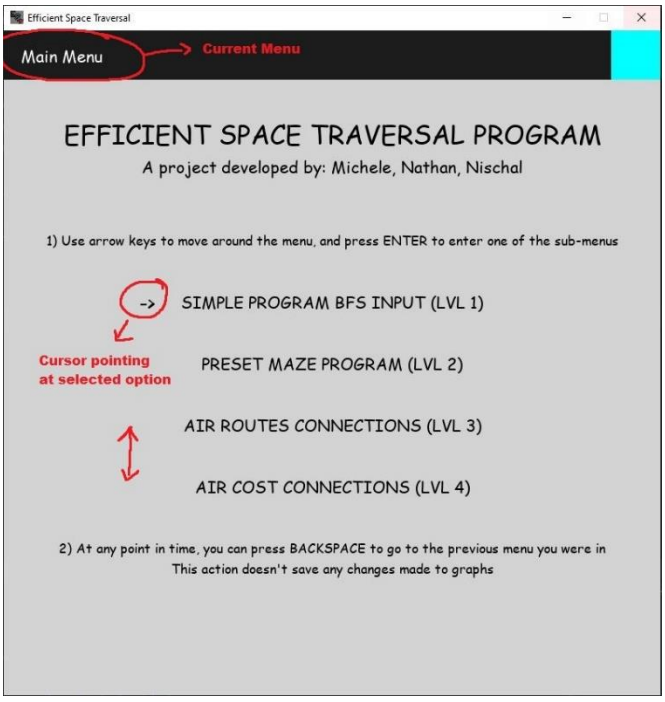
A CSC111 Final Project by:

Michele Massa, Nischal Nair, Nathan Zavys-Cox

Note: This "manual" is to be used as a more careful guide to use our program, and as a proof of how the program should work according to the inputs and outputs displayed.

Prelude: The Main Menu:

In the main menu, the user is able to select which simulation they want to go into (4 different levels). In this menu, the user will have to use the vertical arrow keys to navigate the different options (as pointed by the cursor (→)), and enter the sub-menus related to these simulations by pressing ENTER when hovering over their option of choice.

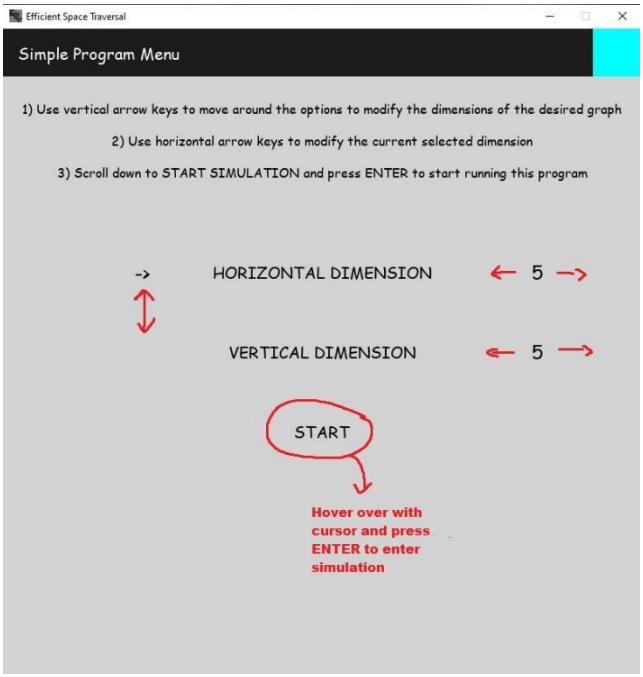


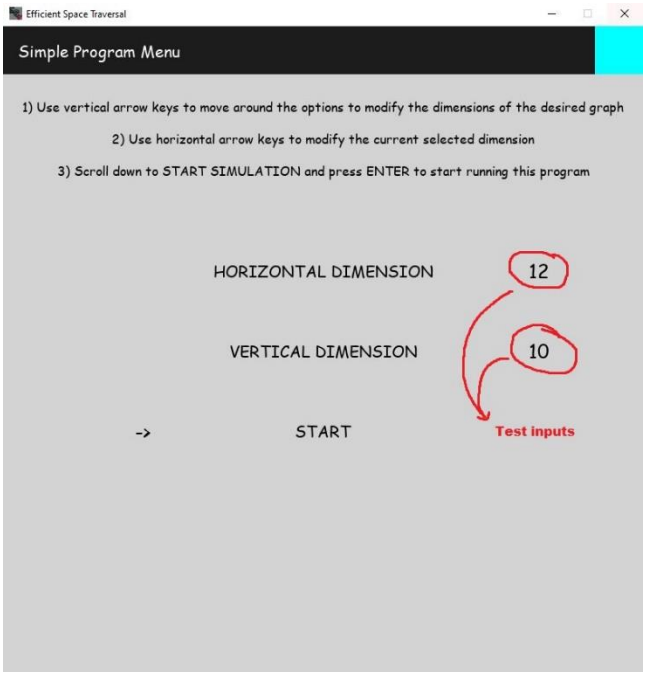
Moving Around Menus:

Apart from being able to enter menus by pressing ENTER and following the instructions on the Pygame window, pressing BACKSPACE takes you back to the previous window. Note: If you have modified the attributes of a graph (grid) and go to the previous menu, the attributes of that graph won't be saved.

Level 1.1: The Simple Program Menu:

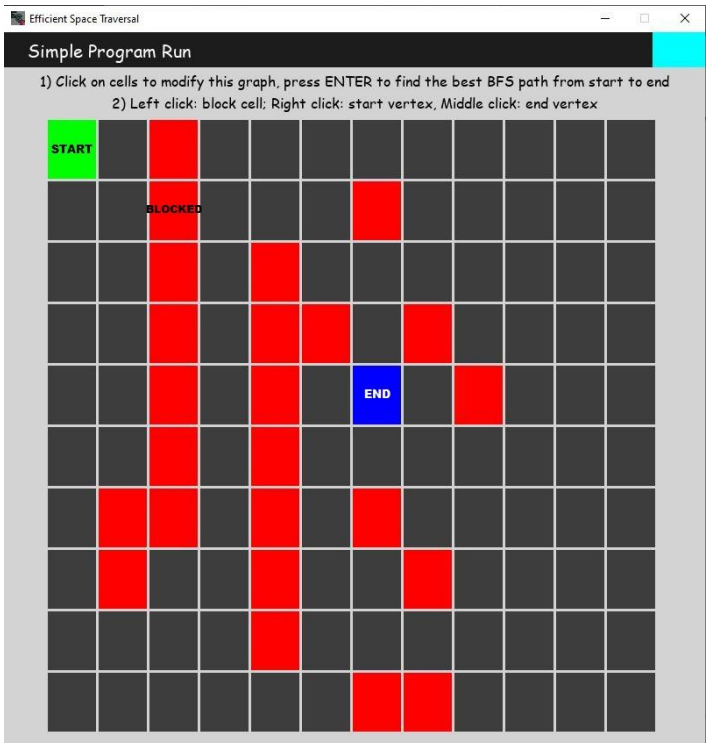
In the simple program menu, the user is able to modify the dimensions of the grid that will represent the space to be traversed in the next window. In this menu, the user will use the vertical arrow keys to navigate the different options (as pointed by the cursor (→)), and modify the dimension attributes by using the horizontal arrow keys when hovering over the desired option. Once ready, the user has to hover over to the START button and press ENTER.



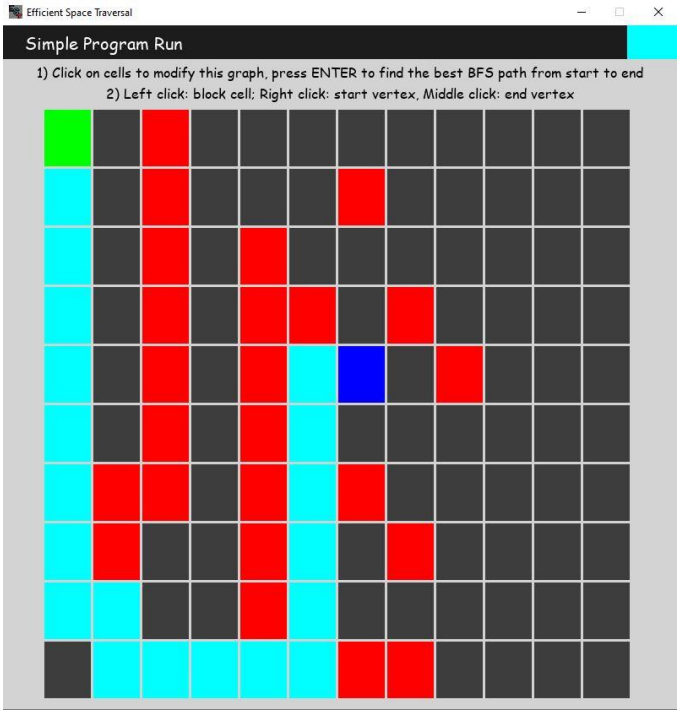


Level 1.2: The Simple Program Run:

In the simple program run window, the user is able to mutate the grid (representing the space to be traversed) by using mouse inputs. Left clicking on a cell BLOCKS that cell (coloured in red), making it non-visitable when calculating a path. Right clicking a cell sets that cell to be the START vertex (only 1 vertex can be the start of a path, coloured in green). Middle clicking a cell sets that cell to be the END vertex (only 1 vertex can be the end of a path, coloured in blue).

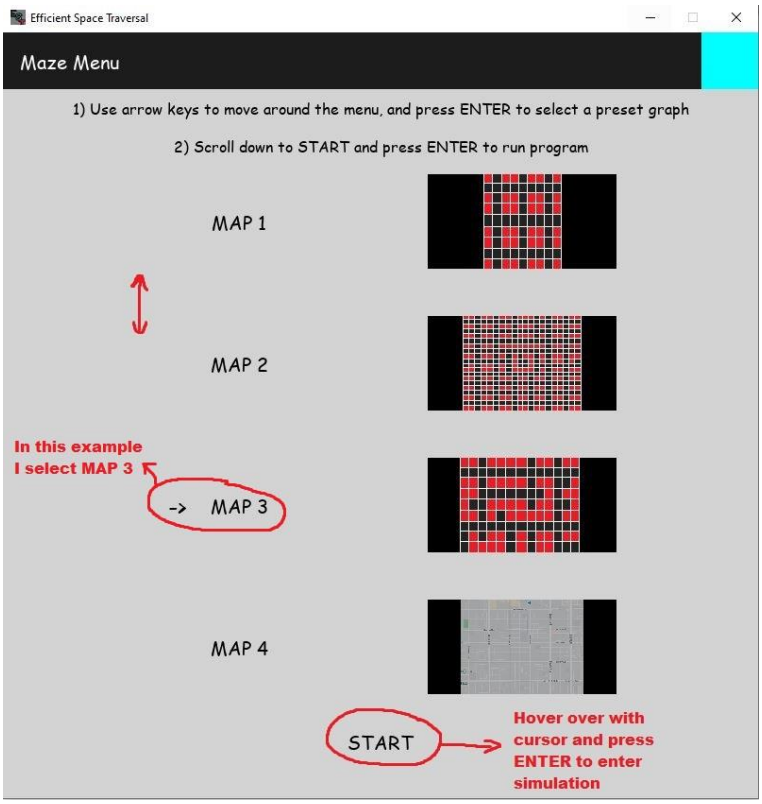


After mutations have been done and the START and END vertices have been set, the user can click ENTER for the program to calculate the shortest path according to the “simpler” Breadth First Search Algorithm, which will be highlighted in cyan.



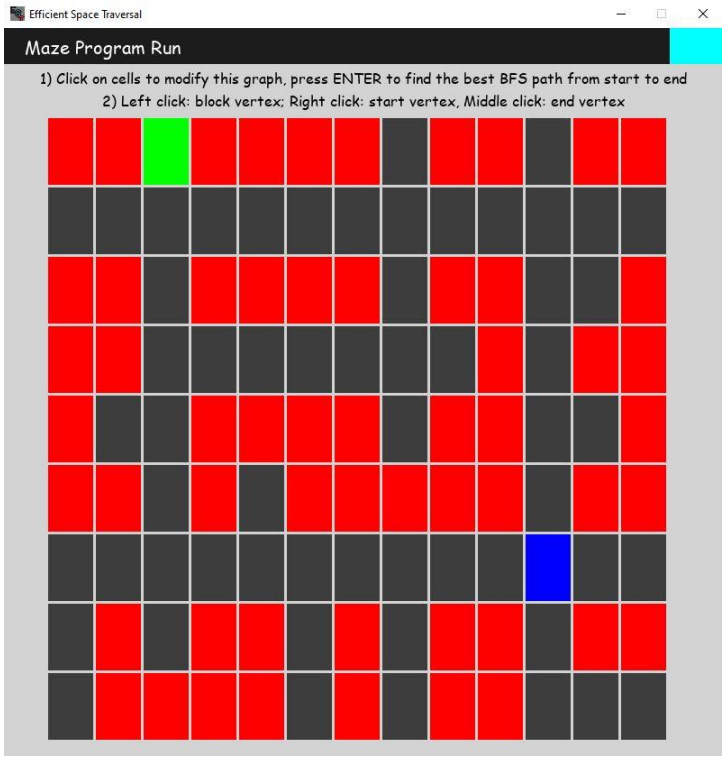
Level 2.1: The Maze Program Menu:

In the simple maze menu, the user is able to select one of the pre-set grids that will represent the space to be traversed in the next window. In this menu, the user will use the vertical arrow keys to navigate the different options (as pointed by the cursor (→)), and click ENTER when hovering over the desired option to select that pre-set. Once a pre-set has been selected, the user has to hover over to the START button and press ENTER (not selecting a pre-set takes the user to the default map = Map 1).

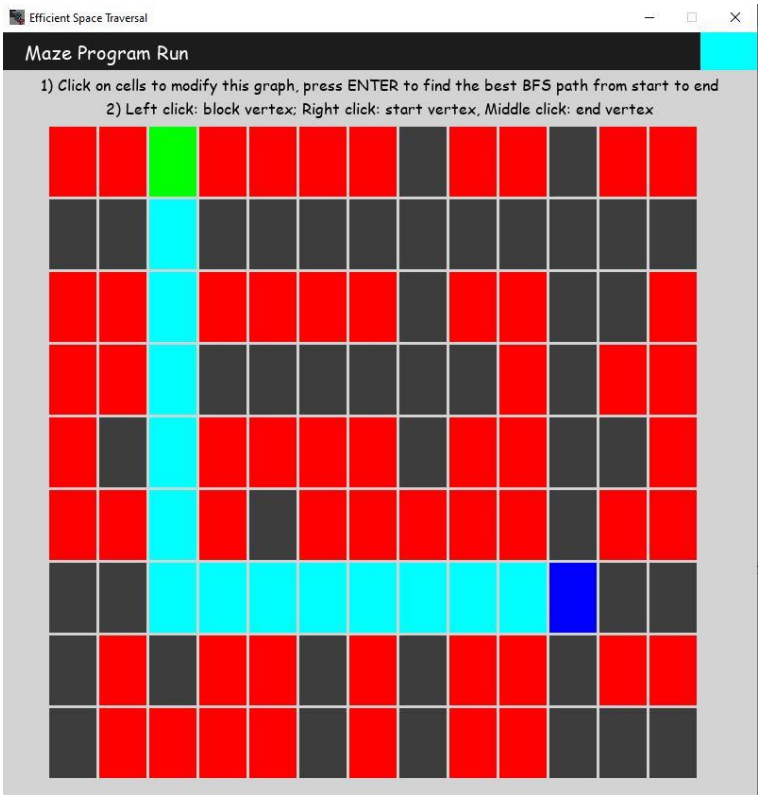


Level 2.2: The Maze Program Run:

In the maze program run window, the user is able to mutate the grid by using mouse inputs., the same way as in step 2.2. Left clicking on a cell BLOCKS that cell (coloured in red), making it non-visitable when calculating a path. Right clicking a cell sets that cell to be the START vertex (only 1 vertex can be the start of a path, coloured in green). Middle clicking a cell sets that cell to be the END vertex (only 1 vertex can be the end of a path, coloured in blue).

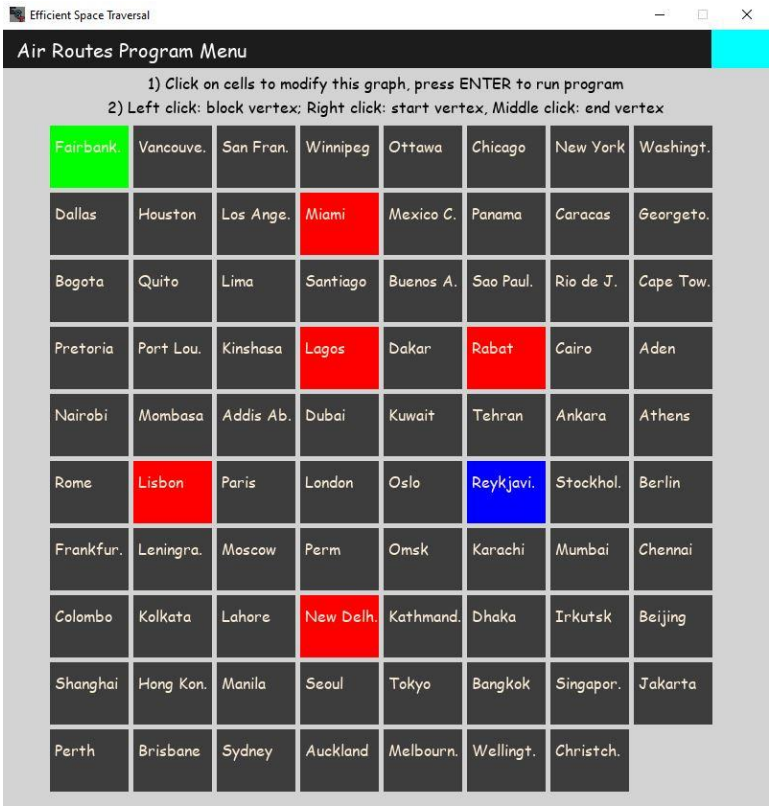


After mutations have been done and the START and END vertices have been set, the user can click ENTER for the program to calculate the shortest path according to the “simpler” Breadth First Search Algorithm, which will be highlighted in cyan.



Level 3.1: The Air Routes Menu:

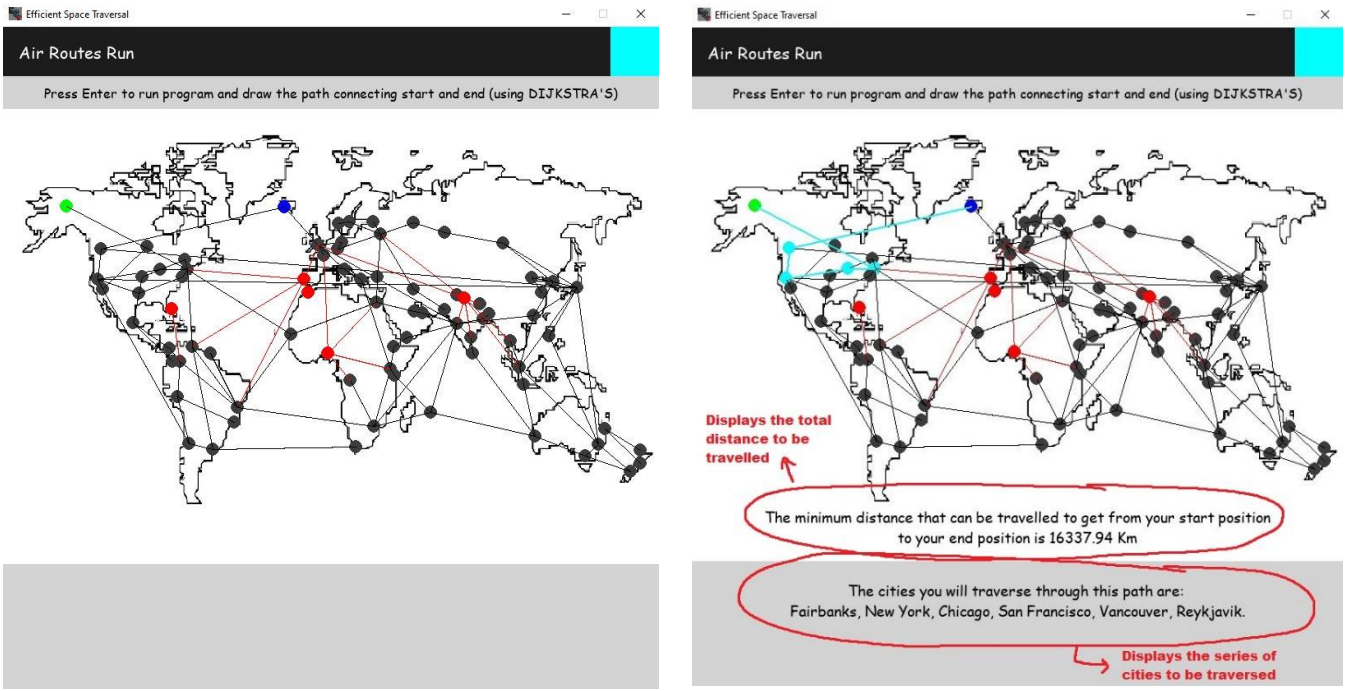
In the air routes program menu, the user is able to modify a grid with the name of the cities that can be traversed in the air routes program run by using mouse inputs the same way as in step 2.2. Left clicking on a cell BLOCKS that cell (coloured in red), making it non-visitable when calculating a path. Right clicking a cell sets that cell to be the START vertex (only 1 vertex can be the start of a path, coloured in green). Middle clicking a cell sets that cell to be the END vertex (only 1 vertex can be the end of a path, coloured in blue). After mutations have been done and the START and END vertices have been set, the user can click ENTER to go into the visualization for this simulation.



Level 3.2: The Air Routes Program Run:

In the air routes program run window, the user is presented with a world graph displaying the air route connections between certain cities (taking into account the mutations preformed on the graph in the air routes program menu window). In this window the user can press ENTER to display the most efficient path going from the START vertex to the END vertex with respect to

the total real-world distance (Using Dijkstra's algorithm to minimize the total distance traversed).



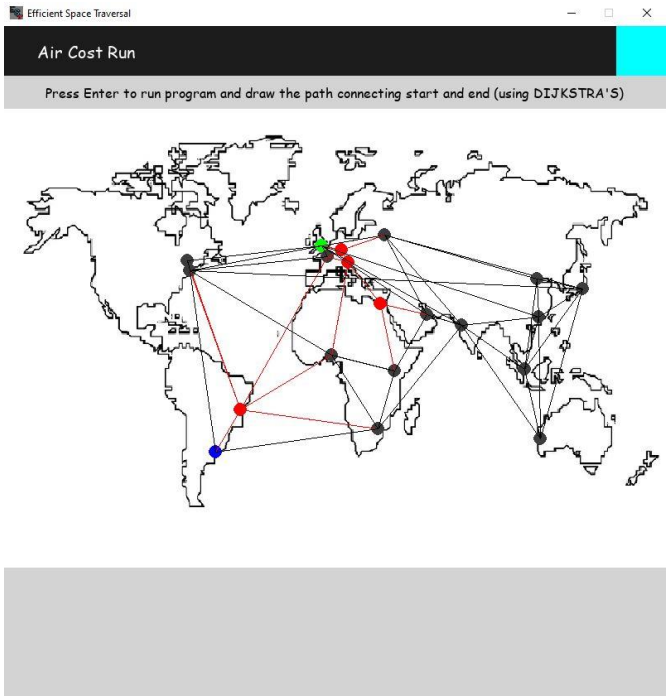
The most efficient path with respect to this metric of weight (total distance) will be highlighted in cyan, and the user will be provided with the specific total distance to be travelled, and a list containing the series of countries (in order) showing the path highlighted in the image.

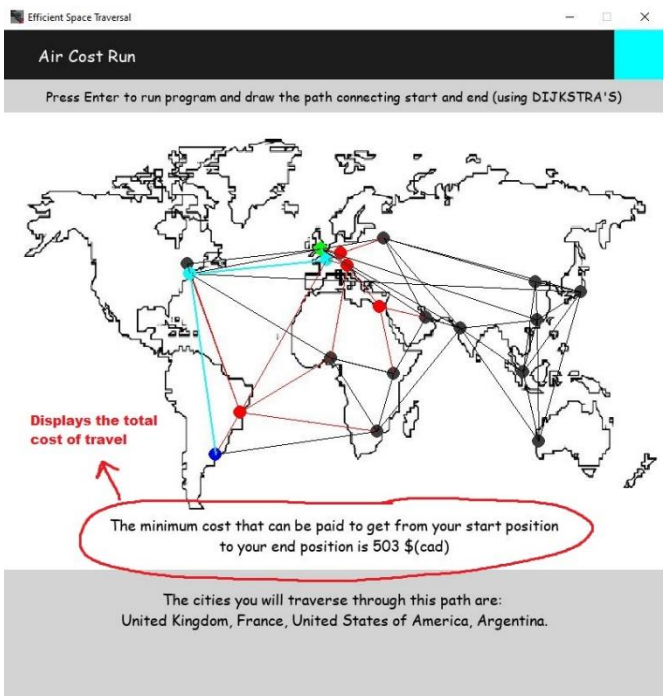
Level 4.1: The Air Cost Menu:

In the air cost program menu, the user is able to modify a grid with the name of the countries that can be traversed in the air cost program run window by using mouse inputs the same way as in step 2.2. Left clicking on a cell BLOCKS that cell (coloured in red), making it non-visitable when calculating a path. Right clicking a cell sets that cell to be the START vertex (only 1 vertex can be the start of a path, coloured in green). Middle clicking a cell sets that cell to be the END vertex (only 1 vertex can be the end of a path, coloured in blue). After mutations have been done and the START and END vertices have been set, the user can click ENTER to go into the visualization for this simulation.

Level 4.2: The Air Cost Program Run:

In the air cost program run window, the user is presented with a world graph displaying the air cost connections between certain countries (taking into account the mutations preformed on the graph in the air cost program menu window). In this window the user can press ENTER to display the most efficient path going from the START vertex to the END vertex with respect to the lowest cost of travel (Using Dijkstra's algorithm to minimize the total cost of a path).

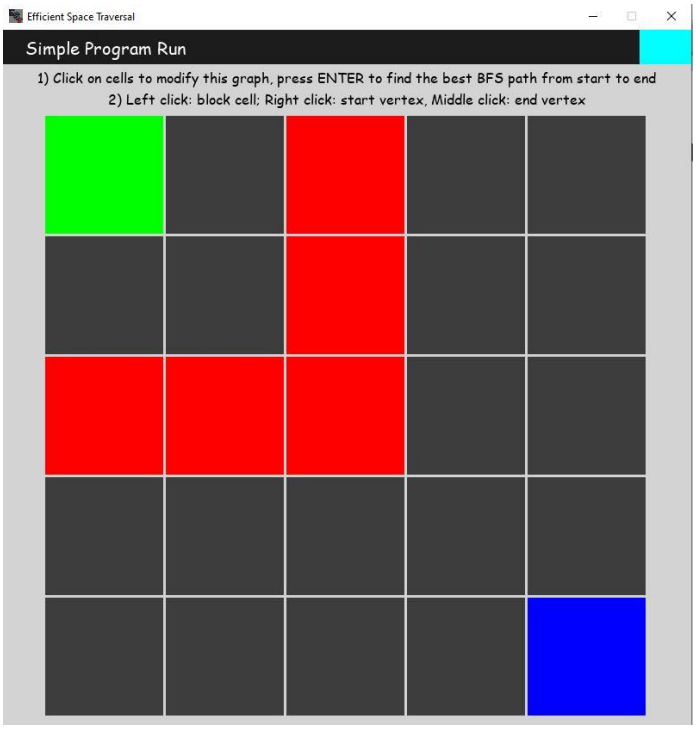


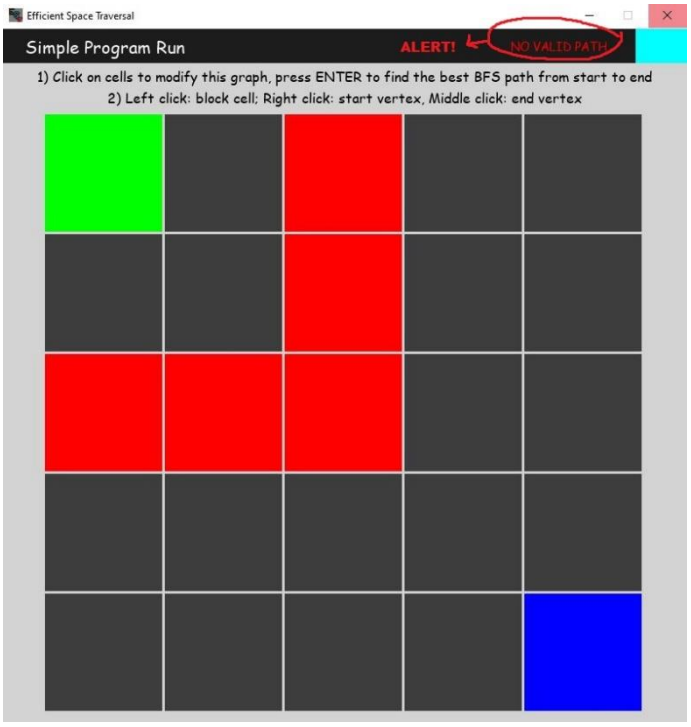


The most efficient path with respect to this metric of weight (total cost) will be highlighted in cyan, and the user will be provided with the specific total cost to be paid, and a list containing the series of countries (in order) showing the path highlighted in the image.

Special Case:

If the user ever enters START and END vertices which are not connected, they will be met with an alert on the top right corner of the window letting them know that there is NO VALID PATH between the vertices they selected.





Lastly: as a general note, specially for the first two levels of implementation, remember that after pressing ENTER in one of the run windows, you should go back to the previous window and restart the graph. Trying to compute new paths doesn't raise an error, and it works but it might lead to the display of more than one path on the same grid at the same time.

THE END