José Eduardo Morales
March 23rd, 2023

Shortest Path Assignment

Introduction

This assignment consists in creating a graph as a ADT in C++ to test the distances to all vertex using Dijktra's algorithm of shortest path. It is implemented using matrices and creating random graphs with densities of 20% and 40% on a graph of 50 nodes with distance range of 1 to 10.

Output of the program

```
117     }
```

Vertex	Dist	Path
n0	0	n0
n1	2	n0 -> n1
n2	5	n0 -> n12 -> n33 -> n9 -> n2
n3	7	n0 -> n12 -> n3
n4	6	n0 -> n12 -> n33 -> n9 -> n2 -> n4
n5	5	n0 -> n12 -> n17 -> n5
n6	9	n0 -> n12 -> n33 -> n29 -> n6
n7	9	n0 -> n16 -> n14 -> n7
n8	5	n0 -> n12 -> n47 -> n8
n9	4	n0 -> n12 -> n33 -> n9
n10	6	n0 -> n12 -> n33 -> n34 -> n10
n11	7	n0 -> n12 -> n33 -> n31 -> n11
n12	1	n0 -> n12
n13	7	n0 -> n1 -> n13
n14	6	n0 -> n16 -> n14
n15	7	n0 -> n1 -> n19 -> n37 -> n15
n16	3	n0 -> n16
n17	4	n0 -> n12 -> n17
n18	4	n0 -> n12 -> n47 -> n18
n19	4	n0 -> n1 -> n19
n20	7	n0 -> n12 -> n17 -> n5 -> n20
n21	6	n0 -> n1 -> n21
n22	6	n0 -> n16 -> n22
n23	8	n0 -> n12 -> n33 -> n34 -> n23
n24	7	n0 -> n12 -> n33 -> n31 -> n24
n25	4	n0 -> n12 -> n33 -> n25
n26	7	n0 -> n12 -> n47 -> n8 -> n26
n27	8	n0 -> n12 -> n47 -> n18 -> n27
n28	4	n0 -> n12 -> n33 -> n28
n29	6	n0 -> n12 -> n33 -> n29
n30	6	n0 -> n30
n31	4	n0 -> n12 -> n33 -> n31
n32	7	n0 -> n12 -> n47 -> n18 -> n32
n33	2	n0 -> n12 -> n33
n34	4	n0 -> n12 -> n33 -> n34
n35	6	n0 -> n1 -> n19 -> n37 -> n35
n36	7	n0 -> n12 -> n17 -> n5 -> n36
n37	5	n0 -> n1 -> n19 -> n37
n38	8	n0 -> n12 -> n47 -> n38
n39	8	n0 -> n1 -> n39
n40	7	n0 -> n16 -> n40
n41	8	n0 -> n12 -> n33 -> n44 -> n41
n42	6	n0 -> n46 -> n42
n43	6	n0 -> n12 -> n43
n44	6	n0 -> n12 -> n33 -> n44
n45	8	n0 -> n12 -> n45
n46	5	n0 -> n46
n47	3	n0 -> n12 -> n47
n48	8	n0 -> n12 -> n17 -> n5 -> n36 -> n48
n49	7	n0 -> n12 -> n33 -> n49

```
Avg. Density = 23.0204%
Avg. Distance= 5.7
new random graph created
```

José Eduardo Morales
March 23rd, 2023

```
new random graph created
DISTANCES FROM n0
Vertex  Dist    Path
n0      0       n0
n1      3       n0 -> n34 -> n1
n2      3       n0 -> n28 -> n2
n3      4       n0 -> n28 -> n23 -> n19 -> n3
n4      4       n0 -> n22 -> n48 -> n4
n5      3       n0 -> n33 -> n5
n6      3       n0 -> n30 -> n6
n7      2       n0 -> n12 -> n7
n8      2       n0 -> n34 -> n8
n9      2       n0 -> n28 -> n9
n10     3       n0 -> n22 -> n48 -> n10
n11     2       n0 -> n12 -> n11
n12     1       n0 -> n12
n13     2       n0 -> n22 -> n13
n14     5       n0 -> n47 -> n26 -> n14
n15     3       n0 -> n28 -> n15
n16     5       n0 -> n16
n17     4       n0 -> n33 -> n17
n18     3       n0 -> n22 -> n42 -> n18
n19     3       n0 -> n28 -> n23 -> n19
n20     4       n0 -> n22 -> n42 -> n20
n21     4       n0 -> n21
n22     1       n0 -> n22
n23     2       n0 -> n28 -> n23
n24     4       n0 -> n34 -> n8 -> n24
n25     3       n0 -> n33 -> n25
n26     4       n0 -> n47 -> n26
n27     5       n0 -> n30 -> n27
n28     1       n0 -> n28
n29     4       n0 -> n22 -> n42 -> n29
n30     2       n0 -> n30
n31     3       n0 -> n28 -> n23 -> n31
n32     4       n0 -> n47 -> n32
n33     2       n0 -> n33
n34     1       n0 -> n34
n35     2       n0 -> n34 -> n35
n36     3       n0 -> n12 -> n36
n37     3       n0 -> n12 -> n11 -> n37
n38     4       n0 -> n28 -> n23 -> n38
n39     3       n0 -> n33 -> n39
n40     3       n0 -> n12 -> n40
n41     4       n0 -> n33 -> n41
n42     2       n0 -> n22 -> n42
n43     4       n0 -> n28 -> n15 -> n43
n44     4       n0 -> n34 -> n35 -> n44
n45     4       n0 -> n12 -> n36 -> n45
n46     5       n0 -> n46
n47     3       n0 -> n47
n48     2       n0 -> n22 -> n48
n49     4       n0 -> n34 -> n49
Avg. Density = 43.102%
Avg. Distance= 3.02
```

José Eduardo Morales
March 23rd, 2023

What I learned

This assignment helped me understand templates, classes, objects and a lot more. I understood how a graph shows a connection between vertex and how to decide on a short route using Dijkstra's algorithm. I also used vectors for the first time, as a more powerful array with verifications on size and function to add or remove members of the vector while also experimenting on c++ arrays. I found how useful are templates for creating function overloads with only one generic type but I stumbled on using arrays while conflicting con char* differences with number. The public and private members of the class where useful to give the user the most user friendly functions. I also learned how to write multiple classes in multiple files, and I created several files. The best part was seeing it all work together.

Code is attached :D!