**Study the Properties of "Small World" and Compare Different Data Structures**

**Due: 1/17 23:59:59**

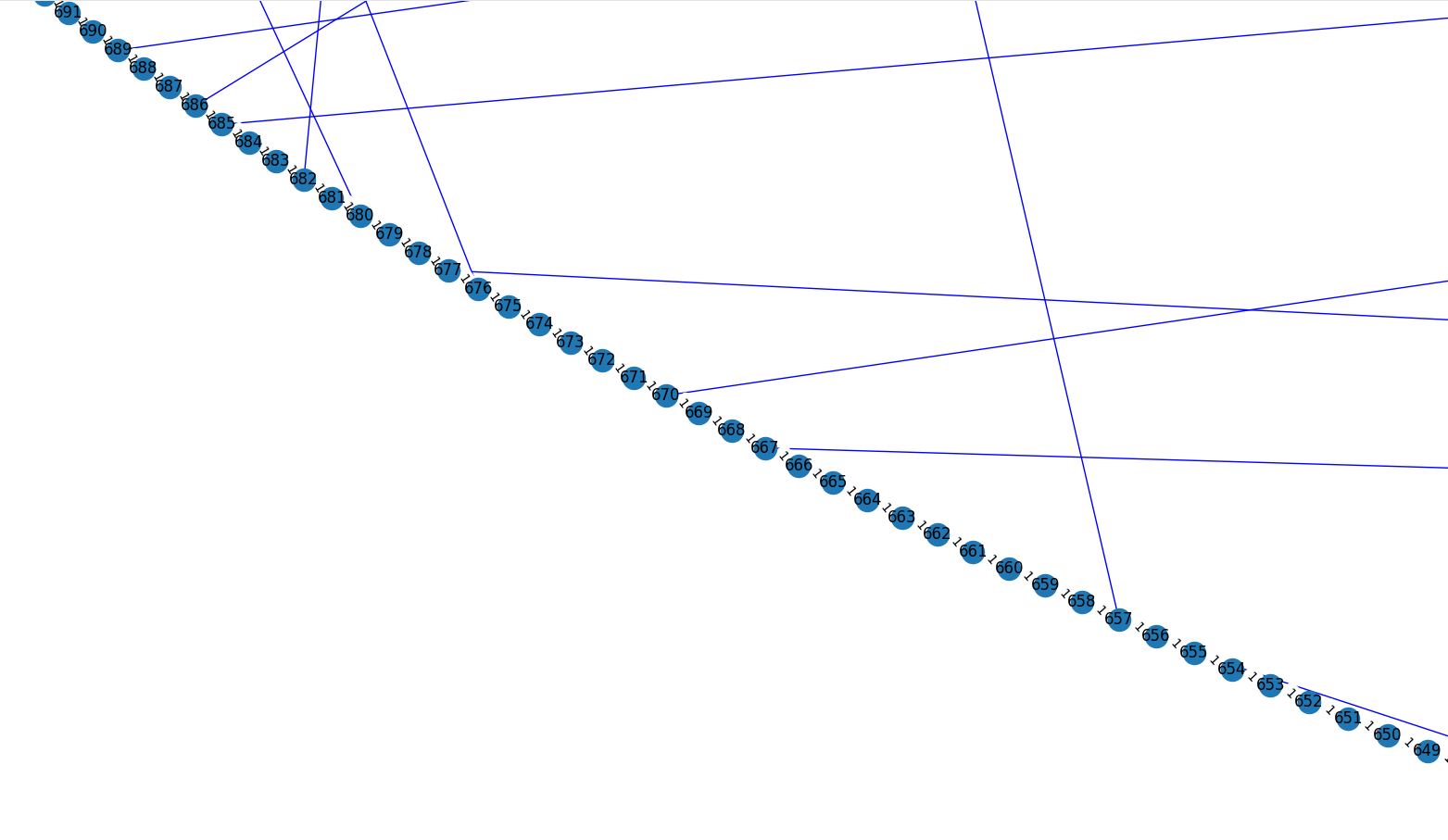
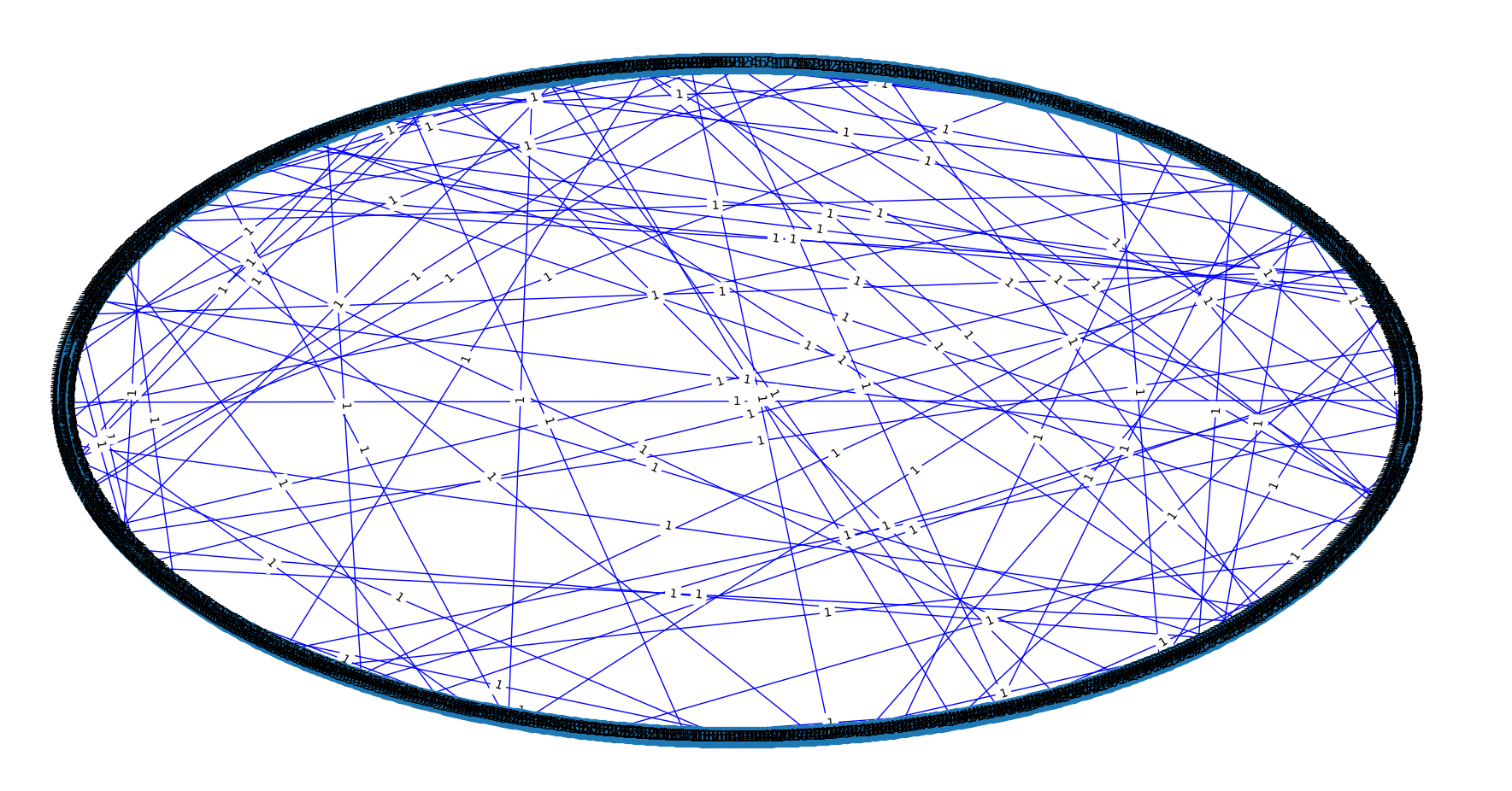
1. Generate a cycle of 1000 nodes. Each edge has length 1.
2. Add 𝑥 random edges. Each random edge has the same length 𝑦.
3. Sample 𝑧 pairs of source and destination, and compute the average shortest distance (𝑑) of these 𝑧 source-destination pairs.

You need to use 2 different structures of heaps. **(In the report, give the name of the data structures that you use, e.g., array and binomial heap).**

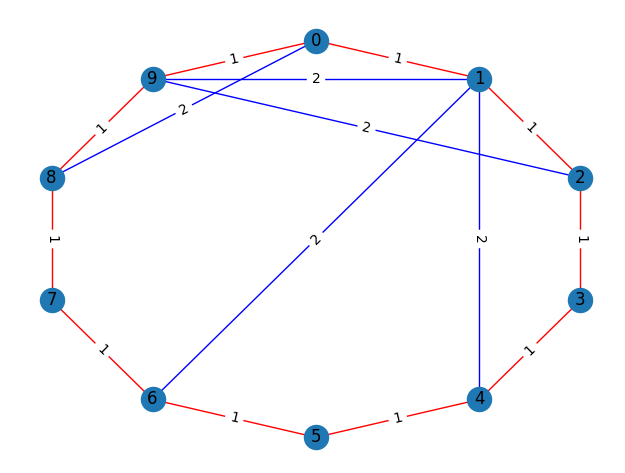
**Your report should contain:**

1. **A picture of the graph with 𝑥 = 100.**

Below graph is a graph with 1000 nodes, x=100 random edges with y=1 length.

****

In order to see more specifier, the graph down below is 10 nodes with x=5 random edges with y=2 length.

****

1. **Responses to the following questions:** 
   1. What is the relationship between 𝑥 and 𝑑?

Below is a graph with 1000 nodes, x random edges with y=1 length, one source vertex = node 0.

Blue curve is dijkstra array, and the orange curve is dijkstra heap.

From the graph we can see that as the x grow, d decrease. This also means that the more edges in a graph, the smaller of the sum of all shortest distance it results. However, the shortest distance when x= 40 and x=50 is bit higher than the trend as I expected. This might be because that the source vertex is not an outlier, if selected other vertexes as source vertex, it would result in a smoother smoother curve.

The curve down below represents the average of ten dijkstra results, finding the shortest path of each graph with 10 different source vertex (source vertex = 0, 1, 2, … ,10). This graph better represents the relationship between x and d. Which is same as I expected. The more x can leads to less d.

The curve below represents the average of 1000 dijkstra results. Both curve become more smoother than further. Still it follows the rule that x and d have reverse relationship.

The intuitive meaning is if there exist short cut from on city to another, then one can choose the short cut. And the city neighbor would also like to go through the short cut.

This experiment also conforms to Six Degrees of Separation. Once a person make friend with another person, the relationship of all people become a lot shorter.

* 1. **What is the relationship between 𝑦 and 𝑑?**

Below is a graph with 1000 nodes, x=100 random edges with y length. The dijkstra source vertex is node 0.

Blue line is dijkstra array, and the orange line is dijkstra heap.

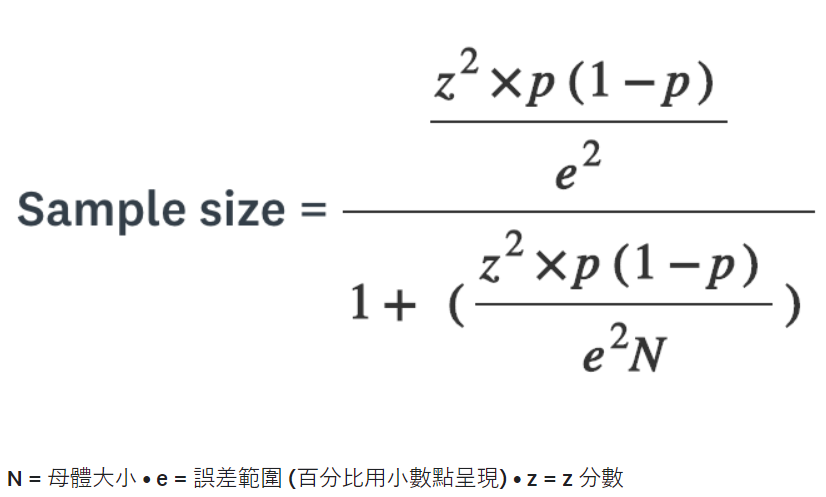
We can see that the distance is up and down, this might because the source vertex I selected can not represent the whole phenomenon.

Below is a graph that represent the average of 1000 experiments.

Form the graph down below, we can say that y and d have a positive incidence, and there are nearly no difference between dijkstra array and dijkstra heap.

This phenomenon is same as intuitive meaning. If the shortcut is longer, the total shortest length is longer.

* 1. **How to choose 𝑧 to obtain a reasonable approximation of the true average shortest distance between all pairs of source and destination?**

In statistic, the best sample size is based on the confidence level and the population size. 

Z-value is a measure of how many standard deviations below or above the population mean a raw score is.



When there are 1000 nodes in the graph, the number of possible pairs is 。Under 95 confidence level and 5% tolerance level, the best sample size under this fomula is 384.

* 1. **Which implementation of Dijkstra's Algorithm is faster?**

Below is a graph when source vertex = 0, with 1000 nodes, x random edges with y=1 length..

Blue line is dijkstra array, and the orange line is dijkstra heap.

Below is a graph of 1000 dijkstra experiments (source vertex = 0, 1, 2, … 1000), with 1000 nodes, x random edges with y=1 length.

Blue line is dijkstra array, and the orange line is dijkstra heap.

From the two graph above, we can see that dijkstra array is slower than dijkstra heap. These is same as

what teacher taught in class. The total time complexity of dijkstra using array is , and the time complexity of dijkstra using heap is n個insert + n個extract min +個decrease key . Since m is not very big here, therefore, the time complexity of dijkstra algorithm using heap is faster than array.

Due to above reason, we can say that as the function of dijkstra heap is nearly the same as dijkstra array but the time is it takes is much smaller. It is better to use dijkstra heap when there are only 1000 nodes in the graph.

1. My code and result data : <https://github.com/yvonne90190/Data-Structure-HW/tree/main/hw4>
2. Even though I spend a long time writing this homework and cannot start my winter vacation. I really like this homework. Unlike the past homeworks, I use Python this time and it is also the first time I print a picture by programming. I spend a long time researching what tool to use and how to use the matplotlib in Python. The time when I finally draw a circle with vertexes, I feel extremely happy.

**\*\* You need to support your answers with experimental results. \*\***

**\*\* You also need to explain how you obtain the results. \*\***