**Manual**

1. **Christofides algorithm**
   1. **Setting up running environment.**
      1. **Python3**

I used Python3 programming language in this project. I suppose you must be familiar with Python, but if not. You can quickly get started with Python and know how to print ‘Hello World’ using python in 10 minutes.

Below are some useful links about download and learning Python:

<https://realpython.com/installing-python/>

<https://www.stavros.io/tutorials/python/>

* + 1. **Other packages**

I didn’t use any other third-party packages in this project, you can run my code correctly after installing Python3 successfully.

* 1. **Test data**

You can change the test data in the bottom of ***ca\_for\_tsp.py*** in the **christofides\_algorithm** directory by altering the value of variable **data**. The shape of **data** is two-dimension list, the length of data indicates how many cities in this TSP problem, each element in data represents the positions of this city in our coordinate. The first number is the x coordinate value, the second number is the y coordinate value.

* 1. **Running code**
     1. **Running in command line**

**If you want to run my code in the command line, just enter to the *1. Christofides Algorithm*** directory by **cd** command, then type command ***python ca\_for\_tsp.py.***

* + 1. **Running in PyCharm or other IDLE.**

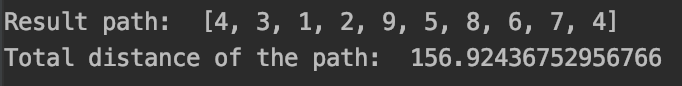
Just run the ***ca\_for\_tsp.py*** python file.

**1.4 Sample Execution:**

**Input:**

data = [[0, 1], [3, 10], [6, 8], [1, 4], [2, 3], [18, 9], [23, 2], [31, 61], [18, 7], [10, 9]]

**Output:**

****

1. **Multiplicative Weight Updates**

I tried, but I don’t know how to do it.

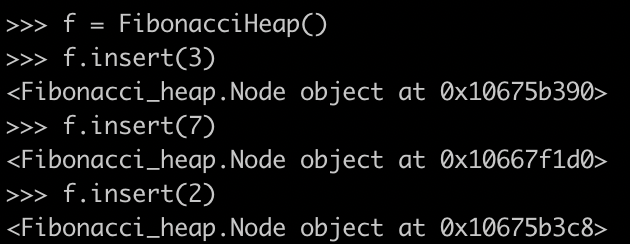
1. **Data Structure**
   1. **Fibonacci Heaps**
      1. **Running Environment:**

Just as same as the environment in question 1. Python3 without other third-party packages.

* + 1. **Run the code:**

1. If you want to run my test data. Just run the ***Fibonacci\_heap.py*** in ***3.Data Structure*** directory. The two methods to execute python file was shown in the manual of question 1.
2. If you want to test yours data, just follow the sample execution below, initialize a FibonacciHeap object, and using insert method to add values then using other methods to test the function of my Fibonacci Heap structure.
   * 1. **Sample Execution:**

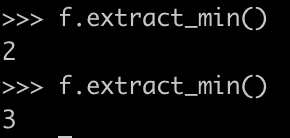
**Insert method:**

****

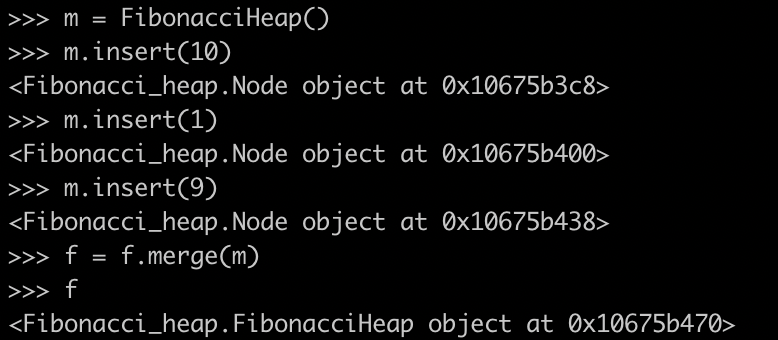
**Get minimum value in heap:**

****

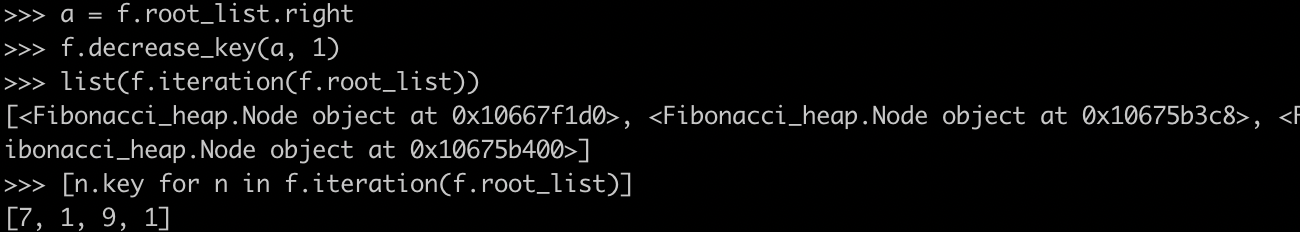
**Extract minimum value in heap:**

****

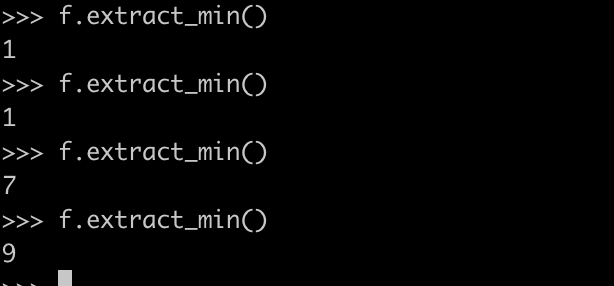
**Merge two heaps:**

****

**Print the values in the new heap:**

****

**Extract minimum value in new heap:**

****

* 1. **Hollow Heaps**

I tried, but I don’t know how to do it.