Wells - Final Document

9th January 2018

Hüseyin Utku ASLAN, Tou AMMAR

USER MANUAL

Giving Input:

To give input, user need to add a text file named “Data.txt” to the folder where executable of the program is located. Text file has the number of wells and houses, and all of their connections. Semicolon will separate data.

There are 2 inputs in the header; number of houses and number of wells. Right format is %;h;**numberOfHouses**;w;**numberOfWells**;

Remaining of the file should contain points, types and id. Correct format is **Type**;**XAxis**;**YAxis**;**Id**;

An example of input file:

```
%;h;6;w;2;
W;29;44;w1;
W;73;80;w2;
H;24;71;h1;
H;56;40;h2;
H;70;2;h3;
H;42;41;h4;
H;6;26;h5;
H;96;5;h6;
```

Getting Output:

After running the program with “Data.txt” in folder, program will generate an output file named “Output.txt”

Output files header is the same as input files header but with addition of total cost. Remaining data shows connections and distance between them. Correct format is **Well_Id**;**House_Id**;**distance**;

An example of output file:
%;h;2;w;2;10;
w1;h1;5;
w2;h2;5

LIST OF MODIFICATIONS

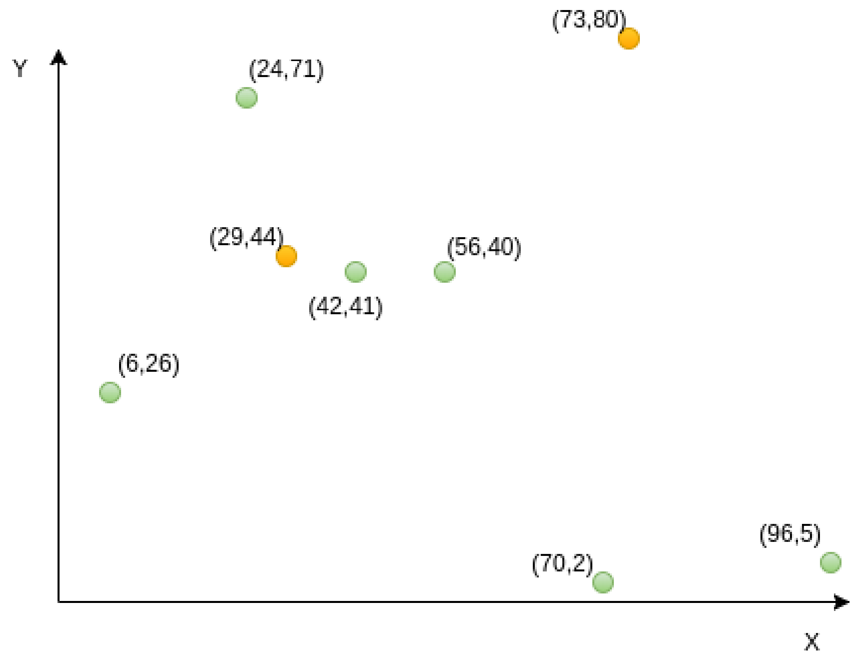
There are no modifications from the last functional document.

TESTS

There were several tests we've made. One of them is the example from functional documents. We have tried different test with inputs generated by our random input generator.

Test from functional document :

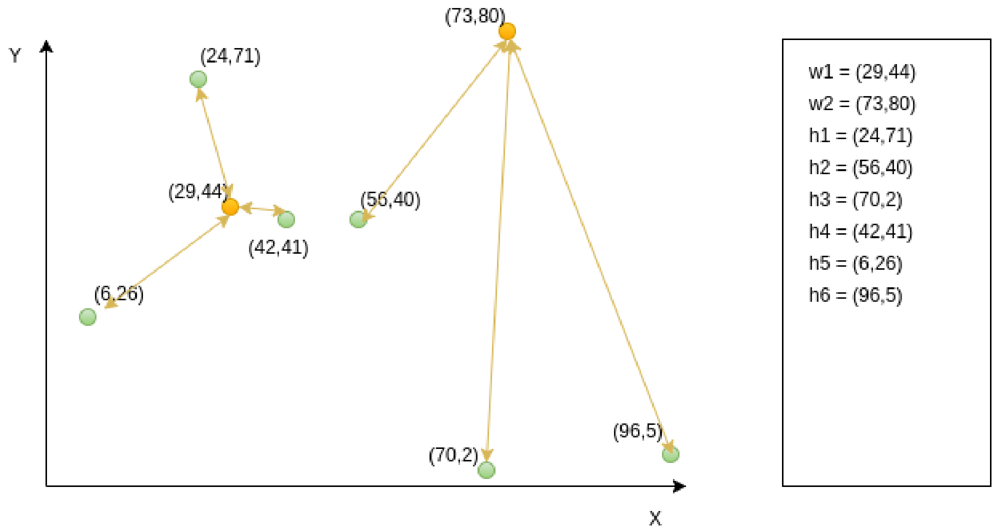
Let w_1 , w_2 ε $wells$ and $h_1, h_2, h_3, h_4, h_4, h_5, h_6$ ε $houses$. Thus every well should have 3 connections.



w1 = (29,44)
w2 = (73,80)
h1 = (24,70)
h2 = (56,40)
h3 = (70,2)
h4 = (42,41)
h5 = (6,24)
h6 = (96,5)

Result:

h_2 , h_3 and h_6 will be connected to w_2 and h_1 , h_4 and h_5 will be connected to w_1 . Total distance is sum(X) + sum(Y) = 68,01 + 199,97 = 267,98 which is minimum distance. It took 0.000318 seconds.



Output.txt :

```
%;h;6;w;2;269.974621
w0;h0;27.459060
w1;h1;43.462628
w1;h2;78.057671
w0;h3;13.341664
w0;h4;29.206164
w1;h5;78.447435
```

Stretch Tests

For 5 wells with 20 connections each, output generated in 0.015039 seconds. Each well is connected to 20 houses. Total number of houses is 100.

For 20 wells with 20 connections each, output generated in 1.19762 seconds. Each well is connected to 20 houses. Total number of houses is 400

For 20 wells with 50 connections each, output generated in 20.094 seconds. Each well is connected to 50 houses. Total number of houses is 1000.

These test durations also includes printing time to the screen.

CONCLUSION

For bipartite matching, Hungarian algorithm gives good results. Hungarian algorithm uses a weighted(distance) matrix that shows distances between each well and house. Hungarian method is good for square matrix and we need to modify our problem according to that. We copied columns for number of connections time. This way we get a square matrix. After getting this weight matrix, we applied hungarian algorithm on it and get print the results.

JOB PARTICIPATION

Hüseyin Utku ASLAN : Research, Programming, Documentation

Tou AMMAR : Research, Programming, Final Build, Testing