IEMS 313
Project Phase 1

Eshan Tarneja, Nikhil Ramesh, Cam Davidson, Kevin Bai

Methodology:

To choose the first machine to place, we created a scoring system for each machine. The machine with the most connections to other machines (forward and backwards), would be placed first in its lowest setup cost block on the factory floor. We did this because we believed if you could place the most connected block first, and build your connections from there, you would be able to minimize transportation costs most effectively. The only exception to this rule would be if two machines had the same amount of connections, as in the example shown to the right. As shown in the example machine one and four both have three connections. So as a tie breaker, we scored blocks based on the cost of their connections, which equates to the sum of the amount of material flow times unit cost per 10m for all connections. For example, in this example machine one would have a score of (5*3) + (1*3) + (4*3) = 30 and machine 5 would have a score of (7*2) + (1*3) + (2*2.5) = 22. The one with the higher score, in this case machine one, would be picked first since this score will eventually be multiplied by the distance of machines from each other, so therefore we would be prioritizing the minimization of connections that were costlier.

To choose the next block, we went through all blocks that were connected to the block we placed first, and in a similar method to our prior scoring, we prioritized the machine whose unit cost per 10 meter times amount of flow to the first machine was highest. Therefore, we would be able to place the costliest machine in terms of transportation cost first and put it in an ideal spot to avoid a large distance. So once we chose this machine, we went through all the remaining open blocks and added together the sum of the setup cost of the machine in that block and the transportation cost times the distance of that block to the first block. After analyzing all the available blocks we would place it in the relative cheapest location. We would utilize this method for all other connections for that first machine.

Once the connections to the first placed block were exhausted, we would basically repeat the process, and find which machine now had the most remaining connections/highest transportation cost and place that on the floor (if it had not already been placed). We would then go through the same process for all the connections to that machine with the same methodology as before. The only difference in the process was that if this new first block hadn't already been placed, instead of placing it simply in an open location that had the lowest setup cost, we would place it in a spot that had the lowest summed setup cost and transportation times cost between the new priority machine and machines that had already been placed which it was connected to. This made sure that the new block wouldn't be placed in a distant location from the existing cluster of blocks just because there was a low setup cost there. This method proceeded in a loop until all connections had been accounted for and all blocks were placed on the floor.

Although likely not optimal, this method is good because it balances the transportation and setup costs. While it would be easy to set machines up based solely on either one of these, almost every time we make a decision in our method it accounts for both, making sure we don't have any machines placements that are driving up our cost too heavily in either category.

Solutions:

Illustration Data

4	10	7	0
8	1	6	9
3	5	0	0
2	0	0	0

final total cost = $2022.00

Data Set 1

```
0  0  0  0
0  0  4  0
0  8  1  2
5  7  6  3
```

final total cost = $7,854.20

Data Set 2

```
 0   0   0   0   0
 0  13   1   0   0
 9   5   2  15   0
 8  12   6  11  16
 4   3   7  14  10
```

final total cost = $45,858.00

Data Set 3

```
 0   0   0   0   0   0   0   0   0   0
 0   0   0   0   0   0   0   0   0   0
 0   0   0   0   0   0  29   0   0   0
 0   0   0   0   0  13  16  37   0   0
 0   0   0   0  28  23   7  19   0   0
 0   0   0  47  11  22  44  46  25  48
 0   0   0  31  42   9  50   2  18  12
 0   0  32   4  27  10  40  45  20   3
38  14  34  43   5  35  21  41   1  15
 6  36  49  39   8  17  30  26  33  24
```

final total cost = $873,640.00

Data Set 4

```
4  5  1  0
3  2  0  0
```

final total cost = $2,829.00

APPENDIX

Illustration Data Example:

The given code reads the file 'Illustration Data.txt' and saves the given information. Once this is run to completion our code sets up a matrix called "floor", which is the size of the specified factory and is what will hold our solution. We also set up a matrix called "finalSUC" which will hold the set up costs for each machine in its specific block. Both are identical at first and set to zeros.

floor and finalSUC

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Our algorithm then creates and fills a matrix from FC called "numarcs" , whose columns refer to machine numbers. The first row in the matrix stores the total amount of connections each machine has and the second row has the total transportation costs(amount of flow times unit cost per 10m) for each of the machine's arcs.

numarcs

3	6	3	3	3	4	4	7	4	3
130	173	170	178	122	194	257	282	219	103

Our code then selects a machine based on which has the most connections(1st row), and if they are tied, by which has the higher transportation cost(2nd row). The chosen machine is referred to as "priority" in the code. So in this case, machine 8 is chosen as "priority". 8 is then placed in the lowest setup cost location.

0	0	0	0
8	0	0	0
0	0	0	0
0	0	0	0

The code then goes on to choose the "next" machine to place based on which machine that is connected to 8 that has the highest transportation cost. In this case it is 4. It then places 4 in the location that minimizes the sum of the setup cost and transportation cost in each available location.

4	0	0	0

8 0 0 0
0 0 0 0
0 0 0 0

This process continues for the rest of connections to 8, except we also consider the cost to other placed machines that the machine is connected to as well.

4 0 0 0
8 0 0 0
3 0 0 0
0 0 0 0

4 0 0 0
8 1 0 0
3 0 0 0
0 0 0 0

4 10 0 0
8 1 0 0
3 0 0 0
0 0 0 0

4 10 0 0
8 1 0 0
3 5 0 0
0 0 0 0

4 10 0 0
8 1 6 0
3 5 0 0
0 0 0 0

4 10 0 0
8 1 6 0
3 5 0 0
2 0 0 0

After all machines connected to 8 are placed, numarcs is updated and a new “priority” is chosen. If the new priority is already on the floor it is still used, just not re-placed. If it is not placed, it is placed in a spot that has the lowest set up cost and transportation cost to machines that are already placed on the floor. The algorithm then runs through all the connections to the new priority and places them if not already placed.

```
4     10    0     0
8     1     6     9
3     5     0     0
2     0     0     0

4     10    7     0
8     1     6     9
3     5     0     0
2     0     0     0
```

This process runs until all connections have been accounted for and all transportation costs/setup costs are calculated. The code then prints out the final floor and cost.

```
floor =
4     10    7     0
8     1     6     9
3     5     0     0
2     0     0     0

final_total_cost = 2022
```

To run this code with a different data set, delete ‘IllustrationData.txt’ from line 3 and replace it with the name of the txt file you would which to run. You then should save and run the file and it should print both the final floor and cost.