stratec biomedical

Internship 2018 - Software

Problem: Pipettor routing

The layout of a medical analyzer platforms pipetting region is split into different zones, for example blood samples, reagents, mixing stations, needle washers, etc. as shown below.

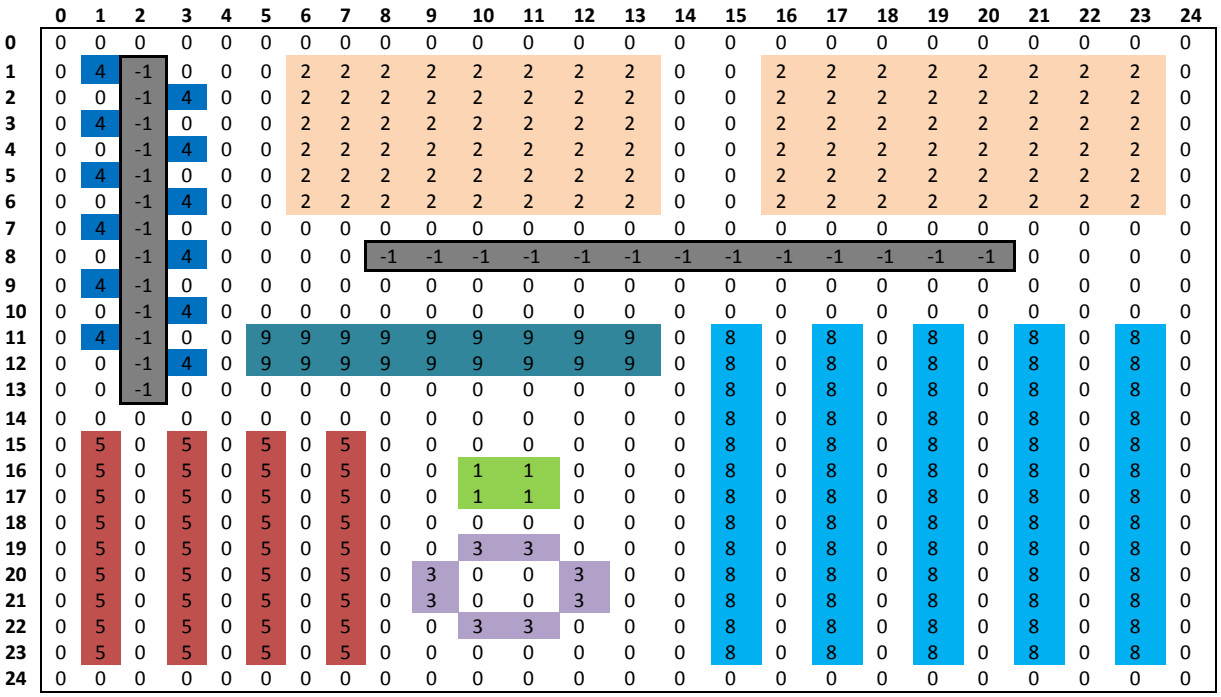
Pipettor systems are often driven at high speeds, pressures and vacuums to maximize the overall throughput (blood samples tested per day). A frequent problem is the possibility of contamination. This can happen when a small droplet forms on the end of the needle due to non-idealized adiabatic effects, which can land and contaminate another patient's blood sample, reagent chemical, empty vials, tubes or other areas of the instrument. In such cases costly and time consuming decontamination procedures have to be performed, not to mention the risk of false positive patient results.

A routing algorithm can avoid going over the more sensitive areas.

stratec biomedical

Modeling

Example:



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
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
1	0	4	-1	0	0	0	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2	2	2	2	0
2	0	0	-1	4	0	0	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2	2	2	2	0
3	0	4	-1	0	0	0	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2	2	2	2	0
4	0	0	-1	4	0	0	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2	2	2	2	0
5	0	4	-1	0	0	0	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2	2	2	2	0
6	0	0	-1	4	0	0	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2	2	2	2	0
7	0	4	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	-1	4	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	0	0
9	0	4	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	-1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	4	-1	0	0	9	9	9	9	9	9	9	9	9	0	8	0	8	0	8	0	8	0	8	0
12	0	0	-1	4	0	9	9	9	9	9	9	9	9	9	0	8	0	8	0	8	0	8	0	8	0
13	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	8	0	8	0	8	0	8	0	8	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	8	0	8	0	8	0	8	0
15	0	5	0	5	0	5	0	5	0	0	0	0	0	0	0	8	0	8	0	8	0	8	0	8	0
16	0	5	0	5	0	5	0	5	0	0	1	1	0	0	0	8	0	8	0	8	0	8	0	8	0
17	0	5	0	5	0	5	0	5	0	0	1	1	0	0	0	8	0	8	0	8	0	8	0	8	0
18	0	5	0	5	0	5	0	5	0	0	0	0	0	0	0	8	0	8	0	8	0	8	0	8	0
19	0	5	0	5	0	5	0	5	0	0	3	3	0	0	0	8	0	8	0	8	0	8	0	8	0
20	0	5	0	5	0	5	0	5	0	3	0	0	3	0	0	8	0	8	0	8	0	8	0	8	0
21	0	5	0	5	0	5	0	5	0	3	0	0	3	0	0	8	0	8	0	8	0	8	0	8	0
22	0	5	0	5	0	5	0	5	0	0	3	3	0	0	0	8	0	8	0	8	0	8	0	8	0
23	0	5	0	5	0	5	0	5	0	0	0	0	0	0	0	8	0	8	0	8	0	8	0	8	0
24	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

For the route “20 1 2 12” (from row 20, column 1 to row 2, column 12), the Pipettor starts aspirating a blood sample from a tall tube, then needs to avoid crossing over other samples (5) or reagents (3, 4, 9 and 8). It cannot cross over walls (-1). It needs to move over the free zones. Mixing containers having height 2 can be crossed freely.

Input text file format

```
<number of rows> <number of columns>
<a[i][j] elements separated by space, rows separated by new line>
<Number of routes to solve>
<Route index> <StartX> <StartY> <DestX> <DestY>
```

Where `a[i][j]` is an integer representing:

- Wall (cannot be crossed), if negative
- Free zone, if 0

- Pipetting area, if positive.
  The positive number represents the height of the respective tube or bottle.
  Mixing station coordinates of height **2** can be passed over.

Output

```
<Solution for route index 1>
<List of motor actions separated by space>
<Solution for route index n>
…
```

Where a motor action is represented by the first letter of the action: **D**own, **U**p, **L**eft, **R**ight.
Example for one route solution:
“

```
3
R U U U U U U U U R R R D D R R R R U U U …
```

”

Problems

a. Solve the problem by computing any route that avoids contamination
b. Solve for the shortest/optimal route

BONUS PROBLEMS:

c. Solve with height map
   Now take into account tube heights. Two new motor actions are available: **Z** for descending the Pipettor, **P** for pulling up by one step. These Z actions can be combined with X-Y actions, e.g. “`ZL ZL ZD PU PU`”, etc. and don’t count as extra routing time. The new challenge is not to break or bend the needle! When the destination or an obstacle is higher, the Pipettor needs to be raised. Upon aspirating/dispensing, the Pipettor height needs to be matched, so not to spill over liquids. The Pipettor starts with the height of the start coordinate position.
d. Solve with acceleration/deceleration.
   A more accurate modeling can take into account acceleration on X-Y-Z directions.
   For every step for which an action is repeated, you can gain 1 speed in that direction. Consider the sequence: “`PL PPLL PPPLLL PPLL PL ZR RR R D`”.
   This models going left over an obstacle while at the same time pulling up the Pipettor, accelerating until speed 3, then decelerating and finally going right, etc.
   At higher speeds more distance can be traveled in the same time (in the above example 2 or 3), but care must be taken not to violate the constraints while “jumping” like this (e.g. crash into a wall or bottle).

e. Come up with your own, more accurate or convenient modeling solution, where all 3 axes move and accelerate independently in the same time. Visualize the steps in a GUI.

Good luck!