X Rastering Cubes

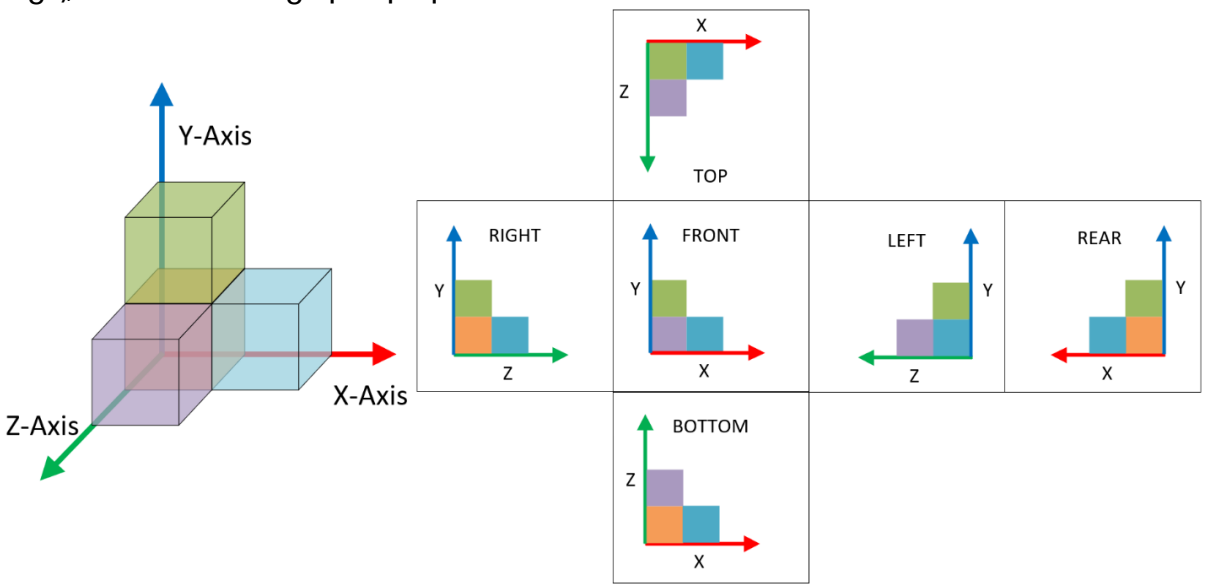
X points

Introduction

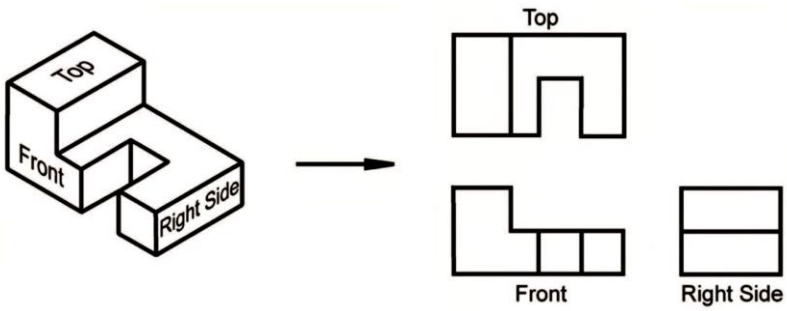
We need a program to draw an orthographic projection of a 3D model (this is the projection of the model in 2 dimensions), which is represented by a set of voxels (a voxel is a cube positioned in the space according to the 3D coordinates (x, y, z)).
The 6 projections are:

View name	Projection axes	View axis
FRONT	xy	+z
REAR	xy	-z
TOP	xz	+y
BOTTOM	xy	-y
LEFT	zy	+x
RIGHT	zy	-x

An example of the 3D model, formed by 4 voxels of different color (the orange one is hidden in the image), and their 6 orthographic projections:



We want to draw only the voxel's edges that are visible and are not touching other edges, as shown in the image below:



Input

The first line indicates the type of projection.

The second line is a positive integer that indicates the number of voxels forming the 3D model.

Finally, the sequence of voxels of the 3D model, each one of them defined by a triplet of "X Y Z" coordinates. Each coordinate is an integer in the range [0, 10].

Output

Voxels are drawn using '+', '-' and '|' symbols.

This is the representation of a single voxel:

```
+-+
| |
+-+
```

The output must be the 2D projection of the input 3D model according to the provided projection type, within a drawing space of 11x11 voxels (see the examples below).

The drawing space is framed by # symbols.

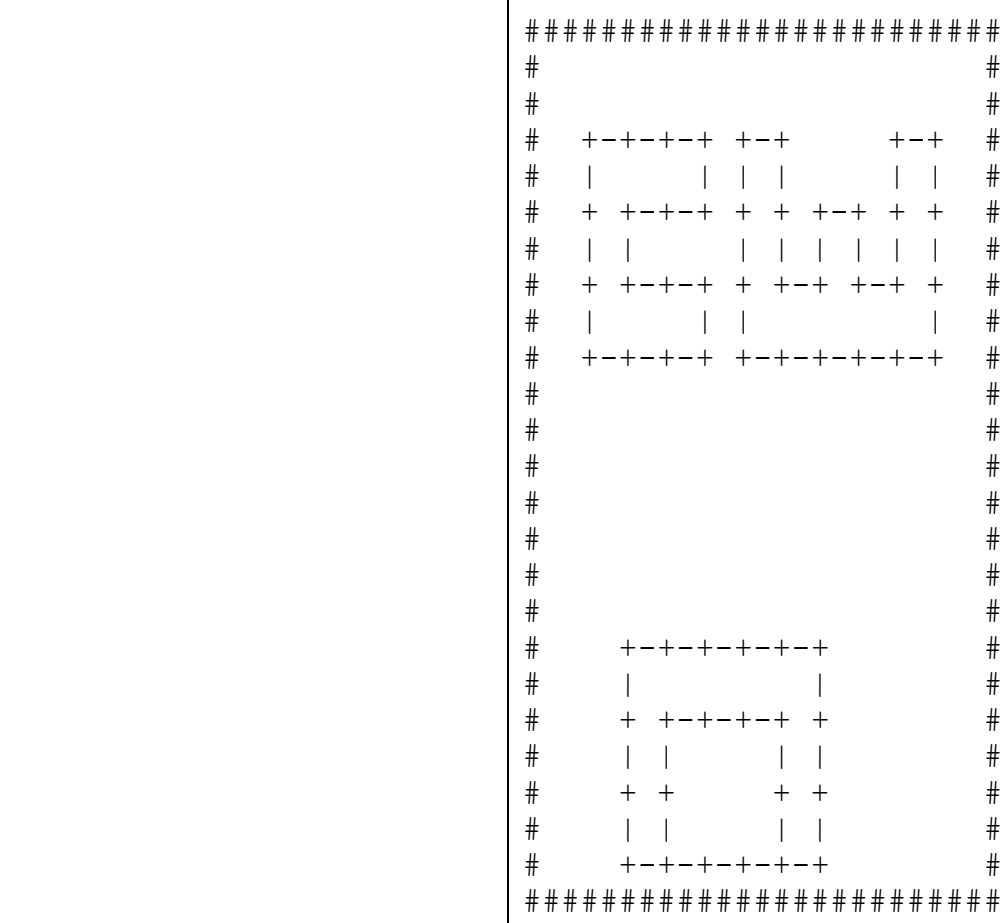
Notice that the origin of coordinates (0, 0) of each 2D projection is a different corner of the drawing space.

Example 1

Input

```
FRONT
38
2 0 3
3 0 3
4 0 3
5 0 3
6 0 3
2 1 3
3 1 3
4 1 3
5 1 3
6 1 3
2 2 3
3 2 3
4 2 3
5 2 3
6 2 3
3 0 4
4 0 4
5 0 4
3 1 4
4 1 4
5 1 4
1 9 0
1 8 0
1 7 0
2 9 0
2 7 0
3 9 0
3 7 0
5 9 0
5 8 0
5 7 0
6 7 0
7 8 0
7 7 0
8 7 0
9 9 0
9 8 0
9 7 0
```

Output



Example 2

Input

```
TOP
13
2 1 5
3 1 5
4 1 5
5 1 5
6 1 5
2 1 6
3 1 6
4 1 6
6 1 6
2 2 5
3 2 5
2 2 6
3 2 6
```

Output

```
##########################
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#     +-+-+-+-+-+        #
#     |   |     |        #
#     +   + +-+ +        #
#     |   | | | |        #
#     +-+-+-+ +-+        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
#                        #
##########################
```