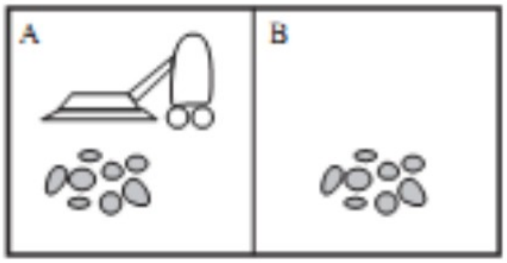
Vaccume Cleaner (Simulation)

Aditya Gupta
1906082

1. (a) WAP to implement a performance-measuring environment simulator for the following vacuum-cleaner world diagram. Your implementation should be modular so that the sensors, actuators, and environment characteristics (size, shape, dirt placement etc.) can be changed easily.



(b) In the addition of the above program (i.e., in 1(a)), WAP to implement a simple reflex agent for the vacuum environment. Run the environment with this agent for all possible initial dirt configurations and agent locations.

Running it:
pip3 install import_ipynb
After downloading the zip q1.zip, we can run cells in `main.ipynb`, or see previous output

Differences from original:
As Sir told we can try different settings, so I found it interesting to try with a 2D array:

a. Size: Instead of giving size directly to a dust object, we assume the quantity of dirt matters, and like again all dirt on a Sofa belongs to that Sofa, so it has More or maybe lesser dirt than other things
b. Shape: This again is specified by object itself, since we can provide different shapes for object in environment.scene
c. Dirt placement: Can be easily controlled by just editting the 'environment.scene' file

Files are:
1. **main.ipynb –** Cells in this notebook can be run for the actual program
2. **environment.scene -**
   It is a file in which we can describe *initial* environment, example file is:

```
-,0,-,0,-,0,-,0,-,0,-,0
+,1,-,0,-,0,-,1,-,0,+,1
+,0,-,0,-,1,],1,+,0,+,1
+,1,-,1,],0,],1,+,0,+,0
+,1,-,1,],0,],1,+,1,+,1
+,0,-,0,+,1,+,0,+,0,+,1
;
-,Sofa
+,Floor
```

```
],Something
```

What this descibes is, the 0 and 1 are whether dirt or not (each coordinate represented by [object,is_dirty] pair),
'-', '+', ']' or **any** other symbol signifies any object, for eg. Here we see that the first row all part of a single object (a Sofa, **logic is that**, if a thing has dirt, **it is better to prefer location on that same object** so cleaning is better rather than unrelated dirt here and there)

After the ';' line, we can specify alternate name for the symbols, for eg. '-' is alias for Sofa. The topmost part of Sofa is all clean, little dirt just below.

Q. What is the performance measure?
It simply always increases, but the more dirt it finds at a region, the more quickly it increases.

Q. How is it modular ?
It all is based on Slide 52 diagram, i found it interesting so tried separated all those components

1. Initial configuration is very much easy to change by just editing text file 'environment.scene'
2. **Sensors**: More sensors can be easily added, as they are separate classes/objects.
For eg, currently using a 'Visual' sensor (to 'see' 2 units farther than current position, what locations has dust),
second is a 'PositionSensor', which is used to tell which position (coord) is the vaccum (technically we can see all the environment but to see only relevant part, albeit a small part, this is needed by cleaner)
3. **Actuators**: Actuators are classes, that have an 'act()' method, it receives a environment matrix, it should return how much coords to move... the internal logic in whatever can be changed as suited, just that it has the 'act()' method, we can pass to **cleaner.attempt_clean**
4. Environment: It is also a different class, should provide a 2D array though so cleaner can find dirt, if needed it can be changed during runtime, or initially with the environment.scene file
5. Not yet done, but grouping of dust based on object, so if Sofa is dirty next time prefer location on Sofa if still dirty: It allows that, and Environment class provides the support, but not needed now

Q. What does your simple reflex agent do ?
Receiving a 2D array of where dirt is, returns a pair of coordinates to how much to move, for eg to move left is to return [-1,0], for right it is [1,0].
Since this is 2D, it can also move diagonally but in limit, ie. 2 unit farther at max
As for how it choses where to move, it **randomly** does so, chosing among the dirty coordinates

PEAS table:

Agent Type	Performance Measure	Environment	Actuators	Sensors
Vacuum Cleaner	How much dirt it found	Floor,objects	Moving,Cleaning	Visual,Position

Originality: I thought it, and at my github: https://github.com/adi-g15/reflex_agents, it actually ended up bit more complex than it should be since i was trying to add it to my simulator project too, ended up making a bit of mess, but should be clean now