

[< Back to Machine Learning Engineer \(Basic\)](#)

Robot Controller

[审阅](#)[代码审阅](#)[HISTORY](#)

Meets Specifications

做得很棒！你有很棒的python基础，项目完成的也很棒，相信你一定能顺利完成接下来的课程！：）

分析模拟环境的数据

任务1: 正确获取模拟环境的长和宽与模拟环境中第3行第6列元素

很棒！用 `len()` 函数可以获得 `list` 元素的个数，通过用索引来访问 `list` 中每一个位置的元素，牢记索引是从 `0` 开始的哦～

任务2: 正确计算模拟环境中，第一行和第三列的障碍物个数。

棒！使用了 `filter` 和 `lambda` 语法完成！

- 还有一种实现方法，可以了解一下：

```
number_of_barriers_row1 = sum([i==2 for i in env_data[0]])
number_of_barriers_col3 = sum([i[2]==2 for i in env_data])
```

任务3: 正确创建并赋值loc_map字典。

做得不错!

建议

用一个函数来实现对 `Start` 和 `Destination` 的坐标获取功能, 使得代码具有更好地复用性。简单来说, 我们希望每一次解决问题时候的“代价”最小, 那么当迷宫矩阵变化的时候, 可以直接通过起点、终点坐标获取函数来获取它们的坐标, 而无需手动修改这里的坐标值。

控制机器人行动

任务4: 正确实现 `is_move_valid_special` 的函数

做得很棒! 也可以参考以下的实现方法, 看看是否会给你一些启发 :P:

```
def is_move_valid_sepcial(loc, act):  
  
    x, y = loc  
    if act == 'u':  
        x -= 1  
    elif act == 'd':  
        x += 1  
    elif act == 'l':  
        y -= 1  
    elif act == 'r':  
        y += 1  
  
    return (0 <= y <= columns - 1) and (0 <= x <= rows - 1) and (env_data  
[x][y] != 2)
```

任务5: 正确实现 `is_move_valid` 函数

任务6: 学生正确回答 `env_data` 这个变量在任务4和任务5中的不同。

任务7: 正确编写一个名为 `valid_actions` 的函数。

任务8: 正确编写 `move_robot` 函数。

- 赞! 试试看使用dict来实现这个功能:

```
def move_robot(loc, act):  
  
    move_dict = {  
        'u': (-1,0),  
        'd': (1,0),  
        'l': (0,-1),  
        'r': (0,1)  
    }  
  
    return loc[0] + move_dict[act][0], loc[1] + move_dict[act][1]
```

任务9: 正确编写 `random_choose_actions` 函数。

(可选) 控制机器人走到终点

(可选) 任务10: 尝试实现一个算法, 能够对给定的模拟环境, 输出机器人的行动策略, 使之能够走到终点。

代码功能性

所有代码均有用且运行时不会产生错误。所给代码足够重现描述的结果。

所有代码均有用且运行时不会产生错误。所给代码足够重现描述的结果。

代码利用函数来避免重复代码。代码包含清晰的注释和变量名称, 可读性较高。

代码的注释中并没有包含对于统计数字和可视化的推理说明。

 下载项目

[返回 PATH](#)

[审阅者 FAQ](#) [Reviewer Agreement](#)