

# Assignment 6 Report

Tan Yee Jian

November 6, 2020

## Contents

<b>1</b>	<b>Question 1: <code>check_valid_move()</code></b>	<b>1</b>
<b>2</b>	<b>Question 2: <code>check_win()</code></b>	<b>1</b>
<b>3</b>	<b>Question 3: <code>pd_map()</code></b>	<b>2</b>

This assignment is well done with a lot of full marks.

## 1 Question 1: `check_valid_move()`

Most got it correct.

1. Some people forgot that the cells in the game are 1-indexed, while Python is 0-indexed. This means you need `game[(inp-1)//3][(inp-1)%3]` to get the cell that the `inp` goes into.
2. Wait, is it `(inp-1)%3` or `inp%3-1`? You need to figure out which one might induce problems for specific inputs. Try taking `inp` as a multiple of 3.
3. Quite a number check through all nine cells, which is fine. But in a generalized version of the game, you will need to check through  $n^2$  cells, which is *very* slow compared to checking **the** cell that `inp` goes into.

## 2 Question 2: `check_win()`

Most got it correct, just check for all the cases of horizontal, vertical, and the two diagonals. Some people split the cases where it is all "X" or all "O", but you can merge these cases by returning any entry in the row/column/diagonal.

### 3 Question 3: `pd_map()`

One of the harder questions, but surprisingly many got it correct; some even had very short solutions. The main idea is to fix a house (by its coordinate), find the **squared distance** to all the shops, and deduce from there. Taking square roots is not necessary, as  $a^2 \geq b^2 \implies |a| \geq |b|$ .