

# Project\_Euler\_015

February 4, 2018

## 1 Project Euler Problem 15

Starting in the top left corner of a  $20 \times 20$  grid, and only being able to move to the right and down, there are exactly 6 routes to the bottom right corner.

How many such routes are there through a  $20 \times 20$  grid?

```
In [7]: # To solve this, we'll create a 21 x 21 array of numbers,
        # initializing every element at 1. For every element along
        # the top row or the leftmost column, there is only one way
        # to reach that point in the grid. For every other point
        # in the grid, the number of paths there is equal to the
        # number of paths to the point just above plus the number
        # of paths to the point just to the left. We can calculate the
        # total number of paths to each point in the grid by applying
        # this sum. The total number of paths will be the value
        # in the lower right corner of this array.

import numpy as np

grid = np.ones((21, 21)).astype(int)
for i in range(1, 21):
    for j in range(1, 21):
        grid[i, j] = grid[i-1, j] + grid[i, j-1]

print("The total number of routes is {}".format(grid[20,20]))
```

The total number of routes is 137846528820.