Project_Euler_018

February 4, 2018

1 Project Euler Problem 18

By starting at the top of the triangle below and moving to adjacent numbers on the row below, the maximum total from top to bottom is 23.

```
3
7 4
2 4 6
8 5 9 3
```

That is, $3 + 7 + 4 + 9 = 23$.

Find the maximum total from top to bottom of the triangle below:

```
75
95 64
17 47 82
18 35 87 10
20 04 82 47 65
19 01 23 75 03 34
88 02 77 73 07 63 67
99 65 04 28 06 16 70 92
41 41 26 56 83 40 80 70 33
41 48 72 33 47 32 37 16 94 29
53 71 44 65 25 43 91 52 97 51 14
70 11 33 28 77 73 17 78 39 68 17 57
91 71 52 38 17 14 91 43 58 50 27 29 48
63 66 04 68 89 53 67 30 73 16 69 87 40 31
04 62 98 27 23 09 70 98 73 93 38 53 60 04 23
```

NOTE: As there are only 16384 routes, it is possible to solve this problem by trying every route. However, Problem 67, is the same challenge with a triangle containing one-hundred rows; it cannot be solved by brute force, and requires a clever method! ;o)

```
In [18]: tri_array = """
         75
         95 64
         17 47 82
```

```
        18 35 87 10
        20 04 82 47 65
        19 01 23 75 03 34
        88 02 77 73 07 63 67
        99 65 04 28 06 16 70 92
        41 41 26 56 83 40 80 70 33
        41 48 72 33 47 32 37 16 94 29
        53 71 44 65 25 43 91 52 97 51 14
        70 11 33 28 77 73 17 78 39 68 17 57
        91 71 52 38 17 14 91 43 58 50 27 29 48
        63 66 04 68 89 53 67 30 73 16 69 87 40 31
        04 62 98 27 23 09 70 98 73 93 38 53 60 04 23
        """

        # This section turns the block of text into a list of lists of numbers.
        # There's one sublist for each row.

        tri_array = tri_array.strip().split("\n")
        for i in range(len(tri_array)):
            tri_array[i] = tri_array[i].split(" ")
        for j in range(len(tri_array)):
            tri_array[j] = list(map(int, tri_array[j]))

        # To solve the actual problem, we add numbers from the bottom up.
        # Starting with the bottom row, each element above it has two
        # options for summing.  Add the element from the bottom to the
        # element from the row above that yields the greater sum.
        # Keep going iteratively up the triangle until you get to the top.
        # The largest sum will show up at the top of the triangle.

        for i in range(len(tri_array)-2, -1, -1):
            for j in range(len(tri_array[i])):
                if (tri_array[i][j] + tri_array[i+1][j] >
                    tri_array[i][j] + tri_array[i+1][j+1]):
                    tri_array[i][j] += tri_array[i+1][j]
                else:
                    tri_array[i][j] += tri_array[i+1][j+1]

        print("The max sum from top to bottom is {}."
              .format(tri_array[0][0]))

The max sum from top to bottom is 1074.
```