

Assignment 3 Report

Tan Yee Jian

October 1, 2020

Contents

1	General	1
2	Question 1 - Anagram	1
3	Question 2,3 - Binomial Coefficients	2
3.1	Iterative	2
3.2	Recursive	2
4	Question 4 - Comparing iterative vs recursive solutions	2
5	Question 5 - Monte Carlo PI	3

1 General

Recursive solutions are usually very easy to write - but inefficient. In some cases, there are ways to optimize recursion (Check out Tail Call Optimized Languages) but in general, it is pretty slow. In this course, use loops whenever you can, but also feel free to use recursion for small cases.

2 Question 1 - Anagram

Most did well. For those who didn't, one logic is as below:

1. Since spaces are not allowed, if the length **s1**, **s2** are different, then they cannot be anagrams of each other.
2. For every letter in **s1**, **s2** should have the same amount. No need to check the other way (every letter in **s2**), since, suppose there is some

letter in `s2` which `s1` doesn't have, but `s2` is already as long as `s1` after this check. Now this extra letters in `s2` will make `s2.len() > s1.len`, a contradiction.

3 Question 2,3 - Binomial Coefficients

3.1 Iterative

Easy to understand and implement, but you MUST use `//` instead of `/` for division then `int()`. Reason being that decimal numbers (which results of `/`) are saved in, in Python, has limited accuracy. If you answer has many digits (say 100++), it can only record up till a certain amount. This will cause your answer to be slightly off once you round it off with `int()`.

On the other hand, if you use `//`, Python can keep track of integers of any(*) length without losing accuracy. Limited by your computer's memory (RAM), that is.

3.2 Recursive

Recursion can be hard to understand, but there are essentially two parts:

1. Base case - to stop infinite recursive calls. In this case, if $\binom{n}{k}$ where $n = k$ or $k = 0$, answer will be 1.
2. Recursive step - break into a "smaller" case. In this case, $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$. An equivalent expression can be written by python functions - essentially calling the same function with reduced parameters.

**

4 Question 4 - Comparing iterative vs recursive solutions

Many of you noticed that recursive solution is slower than iterative solution, and that will get you partial credits. Explanations I accepted include:

- Recursive solutions are easier to think of and express. (YJ: that is true.)
- Recursive solutions take up more memory since many functions are called. (YJ: this is the function call stack being, well, stacked up pretty high.)

- In the recursive solution, one function call leads to two more, and every one of those 2 more - it is increasing exponentially. (YJ: very valid reason. It has a exponential runtime.)

5 Question 5 - Monte Carlo PI

Most get this question, but those who don't did not attempt the question at all. General idea is you have a circle embedded inside a square, and imagine if you throw darts, probability of falling in the circle = area ratio of circle to square. There is a PI there in the ratio, and we can substitute "empirical" probability obtained by simulation to be that ratio, and move terms such that PI is obtained.