ESTRUCTURA DE DATOS 1
Código ST0245

Laboratory practice No. 1: Recursion

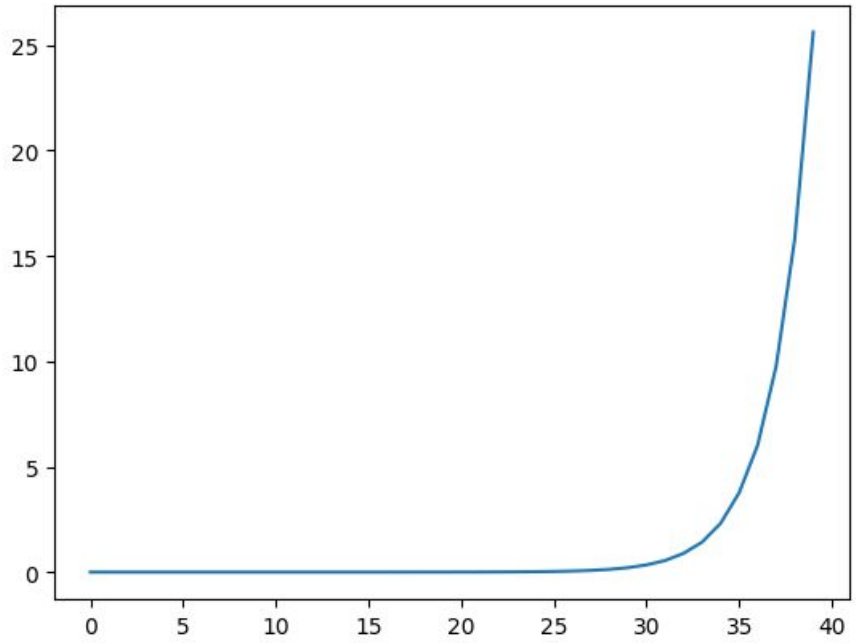
Juan Pablo Ossa Zapata
Universidad Eafit
Medellín, Colombia
jpossaz@eafit.edu.co

Mauricio Jaramillo Uparela
Universidad Eafit
Medellín, Colombia
mjaramillu@eafit.edu.co

1) Practice for final project defense presentation

3.1 Complexity of the board-fill problem: The calculated complexity is described by the equation $T(n) = C_1 + T(n-1) + T(n-2)$. Solving this recursive equation turns out to be somewhat difficult, due to the presence of the n-1 and n-2 parameters. Therefore, this equation was approximated to $T(n) = C_1 2^{(n-1)} + C_1(2^n - 1)$, which is essentially equivalent to $O(2^n)$.

3.2 Plot and time analysis: The following image describes a plot of the time required to compute the result for a given input size:



Given the data, we estimate that it would take around 114688 seconds to compute the result for the input n=50. This time is equivalent to over 31.85 hours.

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

3.3 Would the complexity make this algorithm usable for practical applications of containers that have lengths greater than thousands of centimeters? We consider that the answer to this question depends on the relative size of the objects being transported inside the container itself. For instance, if the container has a length of 2000 centimeters, and the desire is to pack to boxes of 1000 centimeters each, this problem can be simplified to one where the "board" is of size 2x2, which the computer is able to calculate in well under a millisecond. For this reason, we believe that this algorithm could be used for practical applications, as long as the length of the container is 32 times or less that of the items being packed inside, point at which it the execution time approaches unreasonable amounts.

3.4 Explain with your own words how the groupSum5 method works. The exercise groupSum5, asks to indicate if it is possible that given an array of numbers, the sum of a group of them equal to the number called target that is entered as a parameter as well as the arrangement of numbers and the starting point, which is always 0. In addition, this problem has a condition and this is that if a number in any position is divisible by 5, it must be taken into account obligatorily for the sum, also if the next number is a 1, this 1 will not be taken into account for the sum. To solve this problem, a method called skipOne was created which verifies in each position if the previous number was divisible by 5 and if the number in the start position is 1. If so, this method will return a Boolean defined as "false", which will be used in the groupSum5 method to know if the number of the position in start is counted or not. The method groupSum5 will have a base that will return if target is equal to 0 indicating that the sum is possible or not, and will create two branches of recursive form, one where the number of the position start is taken into account in the sum, as long as skipOne is true, and another where the number will not be taken into account as long as this number is not divisible by 5. If any group of numbers is found that, fulfilling the conditions, summed give target, the answer to the exercise will be "true", otherwise it will be "false".

3.5 Complexity of online exercises, numerals 2.1 and 2.1.

Numeral 2.1
Factorial: O(n)
Fibonacci: O(2^n)
Triangle: O(n)
sumDigits: O(log n)
count7: O(log n)

Numeral 2.2
groupSum6: O(2^n)
groupNoAdj: O(2^n)
groupSum5: O(2^n)
groupSumClump: O(2^n)
splitArray: O(2^n)

3.6 Explain the variables with your words.

Numeral 2.1
Factorial: n is the target factorial to obtain.
Fibonacci: n is the target fibonacci to obtain.

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

Triangle: n is the high of the triangle
sumDigits: n is the number of which the digits will be added together
count7: n is the number of which the digits will be added together

Numeral 2.2
groupSum6: n is the size of the array
groupNoAdj: n is the size of the array
groupSum5: n is the size of the array
groupSumClump: n is the size of the array
splitArray: n is the size of the array

4) Practice for midterms

4.1 *Optional*
4.2 *a*
4.3
- *4.3.1 n-a, a, b, c*
- *4.3.2 res,solucionar(n-b,a,b,c)+1*
- *4.3.3 res,solucionar(n-c,a,b,c)+1*

4.4 *Optional*
4.5
- *4.5.1 return n;*
  n-1
  n-2
- *4.5.2 b*

4.6
- *4.6.1 sumaAux(n, i+1)*
- *4.6.2 sumaAux(n, i+1)*

4.7 *Optional*
4.8
- *4.8.1 return 1;*
- *4.8.2 ni + nj*

4.9 *Optional*
4.10 *b*
4.11
- *4.11.1 n-1, lucas(n-2)*
- *4.11.2 c*

4.12
- *4.12.1 sat*
- *4.12.2 Math.max(fi,fj)*
- *4.12.3 sat*

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473