**Algorithmic Methods: Exam Simulation**
Gianmaria Silvello, a.a. 2017/2018

**Exercise 1**
Consider the recursion tree in Figure 1.

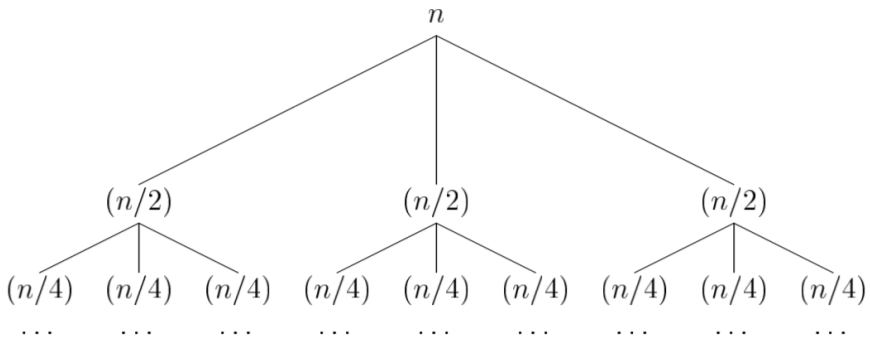


Figure 1: Recursion tree

1. What recurrence relation could generate this recursion tree?
2. How many levels would there be in this tree, as a function on $n$?
3. How many leaves would there be in this tree, as a function of $n$?
4. Solve the recurrence to obtain an asymptotic expression for $T(n)$ as a function of $n$.

**Exercise 2**

- For hashing by the division method, which of the following is the best value for the modulus $m$? Give a brief justification for your answer.
    1. 256
    2. 128
    3. 10
    4. 181
- If the hash table size is 200 and there are 120 elements in the table, what is the load factor?
- Suppose one is using hashing by the division method and the table size $m$ is 53 and the key $k$ is 225. Which bin will this key hash to?

**Exercise 3**

Consider the longest common subsequence dynamic programming algorithm discussed in class.

| | | b | a | a | b | c |
|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 |
| b | 0 | 1\ | 1- | 1- | 1\ | 1- |
| c | 0 | 1\| | 1\| | 1\| | 1\| | 2\ |
| a | 0 | 1\| | 2\ | 2\ | 1\| | 2\| |
| b | 0 | 1\ | 1\| | 2\| | 3\ | 2\| |
| a | 0 | 1\| | 2\ | 2\ | 3\| | 3- |

Fill in the above table with the numbers $c[i, j]$ and the arrows used in the LCS algorithm. Show the longest common subsequence.

**Exercise 4** Answer True or False to the following questions and provide a short justification.

1. Depth First Search (DFS) is a linear time algorithm.
2. There exists a comparison sort of 5 numbers that uses at most 6 comparisons in the worst case.
3. Every computational problem on input size $n$ can be solved by an algorithm with running time polynomial in $n$.
4. The following algorithm to check if the input number $n$ is a prime number runs in polynomial time: For every integer $2 \leq i \leq \sqrt{n}$, check if $i$ divides $n$. If so declare $n$ to be not a prime. If no such $i$ exists, declare $n$ to be a prime.
5. Given $n$ integers $a_1, \ldots, a_n$, the third smallest number amongst $a_1, \ldots, a_n$ can be computed in $O(n)$ time.

**Exercise 5** Suppose that we have numbers between 1 and 100 in a binary search tree and want to search for the number 45. Which (possibly multiple) of the following sequences could be the sequence of nodes examined?

**A.** 5, 2, 1, 10, 39, 34, 77, 63.

**B.** 1, 2, 3, 4, 5, 6, 7, 8.

**C.** 9, 8, 63, 0, 4, 3, 2, 1.

**D.** 8, 7, 6, 5, 4, 3, 2, 1.

**E.** 50, 25, 26, 27, 40, 44, 42.

**F.** 50, 25, 26, 27, 40, 44.

**Exercise 6** Consider the graph in the figure below. Give the sequence of nodes visited by the Breadth-First Search algorithm with source node "A" and draw the resulting Breadth-First tree.

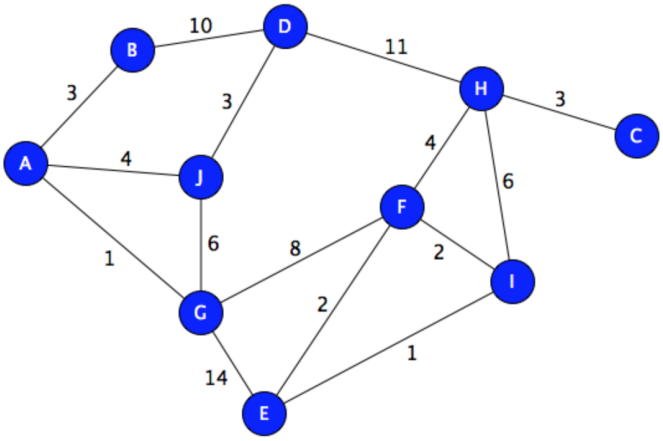


Figure 2: A graph