American University of Armenia, CSE
CS 121 - Data Structures (B)
Homework Assignment 2 Fall 2018

Due Date: Saturday September 29 by 23:59 electronically on moodle

1. Write and test a short recursive method/function in Java or C++ that computes the product of the first k positive integers in an array of integer values without using any loops.

2. Write and test a recursive method/function in Java or C++ that takes input a line of “*” of length 3^k, where k is a nonnegative integer, such that it recursively removes the middle third of each line segment of “*”s. The algorithm stops when each segment consists of exactly one “*”. The figure below shows an output example for $k = 3$.

```
* * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * *                   * * * * * * * * *
* * *       * * *                   * * *       * * *
*   *       *   *                   *   *       *   *
```

3. Consider strings of “A”’s and “B”’s of length n that are built according to the following two rules:

   - for each A already produced, produce an AB
   - for each B already produced, produce a BBA.

   For example, a string of length $n = 13$ that initially starts with A in 4 iterations will result in the following string: $ABBBABBABBAAB$.

   Write and test a recursive method/function in Java or C++ that takes input a string of A’s and B’s and determines if the string was build according to the above two rules. If so, your function should also print the strings build in previous iterations according to the two rules above.

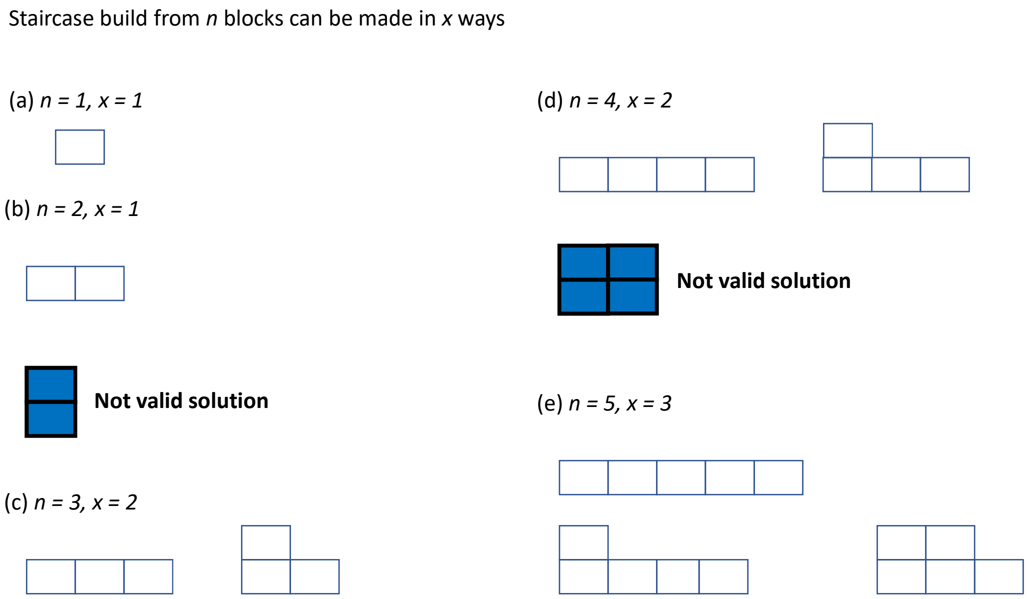
   If the input string to your function is $ABBBABBABBAAB$, then the output of your function should be:

   $ABBBABBABBAAB$
   $ABBBA$
   AB
   A.

4. Given an unsorted array A of integers and an integer x, write a method/function in Java or C++ to find and return a pair of integers a, b in A such that $a + b = x$ (if possible). Otherwise, your function returns `null`. Your function should have $O(n \lg n)$-time complexity.

Part II added

5. Consider building a staircase from a given number of cubic blocks. The figure below shows some examples.



Write a recursive function that takes an integer n, where n designates the total number of blocks used for building the staircase, and returns the number of ways a staircase can built from n blocks.

For example, if $n = 4$, then your output should be "A staircase from 4 blocks can be built in 2 ways".

6. Given a set of strings of equal length implement a hybrid variation of quick-sort and radix-sort to sort the given set of strings.

Your algorithm should start as follows:

Consider the first character of the given strings

- pick a pivot (i.e. the first character of the last string in the set)
- partition the remaining strings by the first character into three sets: those whose corresponding charcter is "less than"; "equal to"; and "greater than" the pivot's character.
- Recursively sort the "less than"; "equal to"; and "greater than" partitions.

With your code submit several test cases. Your test cases should include strings that are the same or similar in writing.