

C S 272/463 Introduction to Data Structures

Lab 9: Recursive Thinking

I. Learning objectives

Objectives 7, 8, 9.

II. Requirements

Implement and test recursive algorithms described below. With the proper use of recursion, none of these methods should require more than a dozen lines of code. Please put all the following function to a file `RecursiveQuestion.java`.

Q1. (15 pts) Write a recursive method to convert a character string of digits to an integer. Example: `convert("1234")` returns 1234. Hint: to convert a character to a number, subtract the ASCII value '0' from the character. For example, if the string `s` has but one character, then the function can return the value `s.charAt(0) - '0'`.

Q2. (15 pts) The Fibonacci numbers are recursively defined as follows:

$F_0 = 0$

$F_1 = 1$

$F_n = F_{(n-1)} + F_{(n-2)}$ for $n > 1$

Please write an algorithm to calculate the k th Fibonacci number F_k using binary recursion.

```
public static int FibBinaryRecursive(int k)
```

Q3. (15 pts) HanoiTower function.

Write an algorithm to solve the Towers of Hanoi problem. (You can take Project 11 at page 450 in your textbook as reference).

Q4. (20 pts) You are given an array of distinct integers, you are required to write a recursive method that prints the permutations of the integers in this array. Write a test function to test your Permutation algorithm. Test your algorithm when the array length is 1, 2, 5, and 10.

Q5. (20 pts) Write a recursive method to solve the project 7 at page 449 in your textbook.

Q6. (15 pts) Rewrite the recursive `pow` method from Figure 8.11 on page 442 so that the time to compute `pow(x, n)` is $\log(n)$. Hint: use the formula $x^{2n} = x^n * x^n$

III. Note

- Specifications for all your classes and methods:

Please properly explain (1) the functionality of the methods, (2) the parameters, (3) the return values, (4) the pre-conditions if there is any;

Please use inline comments, meaningful variable names, indentation, formatting, and whitespace throughout your program to improve its readability.

- You can (but are not required to) design and implement other facilitating methods (E.g., other get and set methods, `toString` method) to finish the implementation of the required methods.

IV. Submission

Submit through canvas a zipped file containing your (1) java file(s) (not .class files).

V. Grading Criteria

1. The score allocation is beside the questions.
2. Please make sure that you test your code thoroughly by considering all possible test cases. Your code may be tested using more test cases.