Data Structures HW1

Logistics

Due date: 4/9 (Sun) 23:59 PM

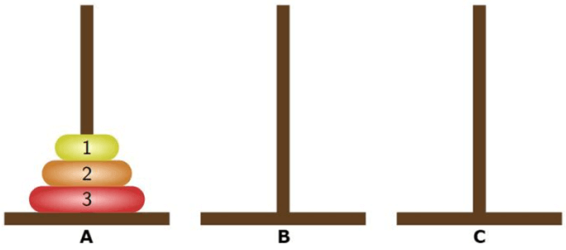
Submission

- Via LMS (no email submission)
- **Two files**

1. output.txt (answer file)
2. Zip file of your program (**Compress your program** as a single file.)

Note: No restrictions on programming languages and platforms.

Problem Description: Tower of Hanoi



In this task, we consider the Tower of Hanoi, a well-known math problem. The problem starts with n disks stacked on the source pole ("A" in the above figure) in ascending order of size and asks how to move them all to the destination pole ("C" in the above). We will basically use the same recursive algorithm that follows all the rules of the Tower of Hanoi problem in the lecture slide, but consider one more factor: "Cost". To do this, we first assign integers from 1 to n to the disks in size order (i.e., assign 1 to the smallest disk and n to the largest one, as shown in the above), and then we assume that the cost of moving each disk varies with size. More specifically, the cost of moving a disk is simply defined as the number assigned to it. For example, moving disk number 3 from one pole (e.g., the largest one on "A" in the above) to the other (i.e., "B" or "C" in the above) costs the value of 3. Your goal is to write a program to measure the total cost of moving the disks in the algorithm. **Don't forget to use RECURSION.**

Example 1. When n=1

You can solve the problem with only a single move (move disk number 1 of cost 1 from the source to the destination.) The total cost is thus 1.

Example 2. When n=3

When n=3, the following moves are required:

Disk 1: 4 times (cost: 1 x 4 = 4)

Disk 2: 2 times (cost: 2 x 2 = 4)

Disk 3: 1 time (cost: 3 x 1 = 3)

Thus, the total cost is 11 (11 = 4+4+3.)

In this task, you will be given an input file (i.e., hw1_input.txt) in the following format:

2
3
1

The number in the first line (i.e., 2) represents the number of cases to be processed, and n (i.e., the number of disks in each case, 1<=n<=20) is given from the next line. For example, the number of disks in the first case is three, and that in the second case is one. For each case, write the result (i.e., the total cost) of each case in "output.txt" in the following format:
11
1

*****Important Note*****
#1. Your program **must include FILE I/O operations** to read from "input_hw1.txt" and write to "output.txt". Otherwise, you cannot receive full marks.

#2. In the output file, **write only one answer per line**. If the format is wrong, the evaluation program might give a wrong score. Below are some wrong examples:

Case 1. Unnecessary information
12345678 Alice Dept. of Mobile Systems Engineering
#1 11
#2 1
Case 2. Multiple answers in a line
11 1

Evaluation Policy (10 pts in total)
Score (10pts) = Implementation (6pts) + Accuracy (4pts)
Accuracy: 4 x # right answers / # total cases
Penalties

1. Unable to build code → Implementation <= 2
2. **Not using recursion → Implementation <= 3**
3. **Plagiarism → Score = 0 (will affect your overall grade)**
4. Late Submission → Score -= 2
5. Wrong output format and missing files (in case you forget to submit output.txt, or…) → Accuracy <= 2