CpSc 2120: Algorithms and Data Structures

Instructor: Dr. Brian Dean   Fall 2020
Webpage: `http://www.cs.clemson.edu/~bcdean/`   MWF 9:05-9:55
Handout 15: Lab #9   Flour 132

1   Recursive Search: Filling Water Jugs

You are standing next to a river with a very content-looking wolf, a head of cabbage, and two water jugs, which have integer sizes $A \leq 1000$ and $B \leq 1000$. In order to boil the cabbage for your dinner, you would like to measure out exactly X units of water. Please write a program that, given A, B, and X, computes how to do this, or determines that the task is impossible. For example:

```
Enter A: 3
Enter B: 4
Enter X: 5


Fill jug 2  [a=0, b=4]
Pour 2->1   [a=3, b=1]
Empty jug 1 [a=0, b=1]
Pour 2->1   [a=1, b=0]
Fill jug 2  [a=1, b=4]


Enter A: 3
Enter B: 6
Enter X: 5


Impossible!
```

To solve this problem, you should use a recursive search through a graph where each node corresponds to a pair of integers (a, b), indicating that you are in the state where jug 1 contains a units of water and jug 2 contains b units of water. You want to start from the state $(0, 0)$ where both jugs are empty, and your goal is to reach some state (a, b) with $a + b = x$. There are 3 possible actions you can take to move between states: filling one of the jugs to its capacity, emptying out one of the jugs, or pouring the contents of one jug into another (until the first becomes empty or the second reaches capacity).

If your program finds a solution, it should print out a step-by-step transcript of the solution such as the one above. Note that you don't need to find the shortest solution — any feasible solution will suffice.

2   Submission and Grading

For this lab, you will receive 8 points for correctness and 2 points for having well-organized, readable code. Zero points will be awarded for code that does not compile, so make sure your code compiles on the lab machines before submitting!

Final submissions are due by 11:59pm on the evening of Friday, November 13. No late submissions will be accepted.