

Project 01: The Two Jugs Problem

Due: March 10 by Midnight

Instructions

For this program you will be designing and implementing an algorithm, in C or C++, to **determine if and how a certain amount of water can be obtained using two jugs. The user should somehow be able to input an amount of water each jug can hold and an amount of water they require.** The program **should output if that requirement can be met using the available jugs, and the steps necessary to do so.**

Also, as a reminder, all of the code for this assignment must be written by you. You may not share code or download solutions off the internet, as doing so will be considered cheating.

Requirements

This assignment has two parts: a design portion and an implementation portion. For the design portion, you must generate documentation describing your program. You might consider including things such as detailed explanations of your various functions and algorithms, how information is stored and is modified, and/or justifications of your design decisions. Diagrams are also an effective way to convey information, provided they also have some description of what is being diagramed. This documentation must be provided along with your program and should be thorough.

Your program must adhere to the following constraints:

- The size of each jug must be a non-zero, positive integer
- The required amount of water must be a non-zero, positive integer
- The required amount of water cannot be greater than the combined sizes of both jugs. It can be less than or equal only
- The required amount of water should be obtainable only by storing water in the jugs. You may not pour water into a final holding area.
- You can only change the amount of water in a jug by:
 - Filling it up to **full from** the water source
 - Completely emptying it back into the water source
 - Emptying from jug A into jug B until jug B is full (A and B can be either jug)
 - Emptying from jug A into jug B until jug A is empty (A and B can be either jug)
- The user can input the requisite information how you wish, either by reading from the keyboard, argument lists, or from a file. If read from a file you should use I/O redirection instead of prompting for a filename.
- The user should receive easy to understand output, as to whether their request is possible, and the steps necessary to achieve it if it is possible
- Must not use STL, with the exception of Strings
- Your code must be well commented
- You must provide a short README.txt file which includes your name and explains how to compile and run your program.
- Additionally, you may write a makefile if you want your code to compile with additional flags.

Submission

A .zip or .rar file containing the following:

1. All files necessary to run your program
2. A README.txt explaining how to compile and run your program
3. Your design document

Rubric

The entire assignment is worth 100 points. The breakdown of those points is as follows.

- 30 points: Design documentation
- 50 points: Code satisfies requirements
- 20 points: Professional coding style
 - 10 points: Adequate comments
 - 5 points: Modularity
 - 5 points: Readability
- If your code fails to compile on the CSE machines you may not receive credit for the programming portion of the assignment. I recommend not making changes to your code without checking for compilation before you submit.