WA3

WEEKLY ASSIGNMENT 3
BME 121, FALL 2016

Due Fri Sep 30 at 4:30 pm

ASSIGNMENT

THE 711 PROBLEM

The following description is taken from Gries, D., "The 711 Problem," Technical Report TR 82-493, Department of Computer Science, Cornell University, Ithaca, New York, May 1982.

> The United States is filled with small grocery stores that, for convenience, are open at all hours of the day and night. In the south, these are called 711 stores, because originally they were open from seven in the morning until eleven at night.
>
> One day, a customer bought 4 items at a 711 store. The cashier bagged them and said, "That will be $7.11 please." The customer asked, "Is it $7.11 because this is a 711 store?" "No," replied the cashier, "I multiplied the prices together and got $7.11." "But you're supposed to add them, not multiply them," said the customer. "Oh, you're right!" exclaimed the cashier. "Let me recalculate … that will be $7.11."
>
> What were the prices of the 4 items?

You can check that the four numbers 3.16, 1.50, 1.25, and 1.20 both add and multiply to a result of 7.11.

BRUTE-FORCE SOLUTION

Although the prices look like real numbers, this is actually an integer problem since all the prices are an integer number of cents. Given an integer n, we are looking for four nonnegative integers a, b, c, d such that a/100, b/100, c/100, d/100 both add and multiply together to form n/100. Multiplying by 100 as needed we can express this another way. Given a nonnegative integer n, we are looking for four nonnegative integers a, b, c, d such that a + b + c + d = n and a * b * c * d = n * 100 * 100 * 100.

The 711 problem has n = 711 and is solved by a = 316, b = 150, c = 125, and d = 120.

It should be clear that we can always arrange the four integers a, b, c, d such that a ≥ b ≥ c ≥ d. By doing so, we eliminate multiple solutions which differ only in the order in which a, b, c, d are stated.

Consider the following brute-force solution. Given n, check every value of a between 0 and n. For each a, check every value of b between 0 and a. For each a and b, check every value of c between 0 and b. For each a, b, and c, use d = n - a - b - c (forcing a, b, c, d to sum to n). If d is not greater than c and the product of a, b, c, d is n * 100 * 100 * 100, we have found a solution with a ≥ b ≥ c ≥ d.

Write a program which will find all a, b, c, d values which solve the 711-style problem for n ranging from 0 to 1000.

Note that there may be multiple solutions for some values of n. For example, the n = 714 case is solved either by a = 250, b = 250, c = 112, d = 102 or by a = 320, b = 150, c = 125, d = 119.

Your program should produce output as shown by the executable solution in the SampleSolution.zip file. Download and unzip the file. Open a command prompt in the resulting folder and run either "dotnet .\wa3.dll" (Windows) or "dotnet ./wa3.dll" (Linux/Mac) to run the sample solution. Note that, because this is a very inefficient search, it slows down signficantly as the value of n increases. Also, there is a long gap between n = 0 and n = 644 with no solutions. The output of this program is showing n, a, b, c, d each divided by 100 (i.e., in the form of the original 711 problem).

In this brute-force search, some numbers will overflow a 32-bit integer representation (C# int). Thus, you should use 64-bit integers throughout (C# long).

SUBMISSION

Name your C# program file as wa3.cs. Use Bme121.Wa3 as your namespace identifier. Include the standard doc-comment block. Submit wa3.cs at the following url.

https://georgefreeman.ca/fileuploader