CSCI2243-Assignment 8

Assigned November 20 due Friday, December 1

Read Chapter 7—you can skip the stuff at the end about Public-key cryptography and the RSA algorithm (though it is the most interesting application and, to some extent, justifies the work done in studying the speed of some some of the basic algorithms). All of the assignment problems below, except for the second extra-credit problem, are taken from the end of the chapter. There are a lot of questions, but the majority are pretty short, asking you to apply basic algorithms using integer arithmetic. Note that the number representations asked for in problem 10 can be computed quickly using the code for the `digits` function given in the textbook, but I want you, just this once, to show the calculation by hand.

Problems 6, 21, and 27 are a matched set: In Problem 21 you are asked to prove that the property holds without using unique factorization into primes–not hard, but a little puzzle. In Problem 27 you are asked to prove the same property, but now you can use unique factorization into primes and the result is much clearer.

Problems 15, and the additional problem, are extra credit You can hand them in any time during the semester. (Prepare them carefully, with complete explanations, and, in the case of part (c) of Problem 15, a properly-written proof.)

1,4,5,6,8,18,21,27,32,33,34.

Extra credit. Problem 15

Additional extra credit problem. This was inspired by a question asked in class. Suppose your country's currency has only two denominations: m dollars, and n dollars, where $\gcd(m, n) = 1$. Since we know that there are integers a, b such that $am + bn = 1$, we can conduct a transaction for any integer number of dollars: Suppose a is positive. In order to pay exactly one dollar, the buyer gives the seller a m-dollar bills, and receives $(-b)$ n dollar bills in change. The same argument works, starting with n-dollar bills, if b is positive. We can replace a and b by ca and cb to conduct a transaction for c dollars.

Earlier, when we were studying mathematical induction, we showed in a specific case (4- and a 7-dollar bills, I think) that we could actually make any *sufficiently large* positive number k of dollars and not need to get change: That is, there exist $a, b \geq 0$ such that $4a + 7b = k$. But we had to use trial and error to find how large k needed to be for this to work.

Show that this works in general: as long as $k \geq mn$, then there exist $a, b \geq 0$ such that $am + bn = k$. Your argument should describe an algorithm, based on Euclid's algorithm, for computing a and b. Illustrate the procedure in the case when $m = 7$, $n = 11$, and $k = 100$.