University of Nevada, Las Vegas Computer Science 477/677 Fall 2015

Assignment 10: Due November 24, 2015

Name:____

You are permitted to work in groups, get help from others, read books, and use the internet. But the handwriting on this document must be your own. You may attach extra sheets, using a stapler.

1. Given items whose weights are 5, 15, 11, 8, 17, and 7, is there a set of those items whose weight is 21?

   Use the pseudo-polynomial algorithm for the knapsack problem shown in class, and show your work.

2. Give a definition of each of these problems, and state what is known about its computational class, by assigning it one of the following phrases.

   (a) It is in the class $\mathcal{P}$-TIME.

   (b) It is $\mathcal{NP}$-complete.

   (c) It is in the class $\mathcal{NP}$-TIME, but no one knows whether it is in the class $\mathcal{P}$-TIME, and no one knows whether it is $\mathcal{NP}$-complete.

   (d) It is not in the class $\mathcal{NP}$-TIME.

   - Independent set.

   - Primality.

   - The halting problem.

3. The Greek mathematician Eratosthenes of Cyrene, who lived during the third and second centuries B.C., is said to have invented the Sieve of Eratosthenes, which can be used to design an algorithm for the primality problem. Is that algorithm a polynomial time algorithm?

4. In order to encode, or decode, a message using the RSA cryptosystem, you need three integers:

   (a) The *modulus* n, which is publicly known large integer, with typically hundreds of bits. Typically, $n = pq$, where p and q are large primes which are kept secret.

   (b) The *message* m, which is a number in the range $1 \ldots n - 1$.

   (c) The *exponent* e, a positive integer, which could be fairly large as well, but never more than $n - 1$.

   The message is encoded by computing $c = m^e (\text{mod } n)$.

   The message is then decoded by computing $m = c^d (\text{mod } n)$, where d is can be computed from e, p, and q, and is secret since p and q are secret.

   Here is some pseudocode which computes $c = m^e (\text{mod } n)$.

   Error Corrected `Fri Nov 20 05:58:33 PST 2015`

```
a = m;
b = e;
c = 1;
while (b > 0){
  if(b is odd)
    c = c*a % n;
  a = a*a % n;
  b = b/2; // Truncated division, as in C++
}
// The output is c.
```

   (a) Is the code above in the polynomial class $\mathcal{P}$-TIME? Explain your answer.

   (b) The code contains a loop. Write the loop invariant for that loop.