**HW**

1. Is the following Bare Bones program self-terminating? Explain your answer.

incr X

decr Y

Yes. In fact, this program halts regardless of the initial values of its variables, and therefore it must halt if its variables are initialized to the program’s encoded representation

1. Is the following Bare Bones program self-terminating? Explain your answer.

Y=X

incr Y

incr Y

while X not 0:

decr X

decr X

decr Y

decr Y

decr Y

while Y not 0:

If X is even to begin with, the program will not terminate. This is because when the first while loop exits, X is 0, but Y is 2. So the second while loop is an infinite loop.

If X is odd to begin with, the program will terminate. This is because when the first while loop exits, X is 0, but Y is 1. So the second while loop exits immediately.

**HW**

1. Suppose a problem can be solved by an algorithm in O(n2) as well as another algorithm in O(2n). Will one algorithm always outperform the other?

No. As a general rule, the algorithm in O(n2) will outperform the one in O(2n) for sufficiently large n, but for small n an exponential algorithm can outperform a polynomial algorithm.

1. Give an example of a polynomial problem. Give an example of a nonpolynomial problem. Give an example of an NP problem that as yet has not been shown to be a polynomial problem.

Within the class of polynomial problems is the sorting problem, which can be solved by polynomial algorithms such as the insertion sort.

Within the class of nonpolynomial problems is the task of listing all the subcommittees that could be formed from a given parent committee.

The Traveling Salesman problem is an example of an NP problem that has not been shown to be a polynomial problem.

1. If the time complexity of algorithm X is greater than that of algorithm Y, is algorithm X necessarily harder to understand than algorithm Y? Explain your answer.

No. Time complexity refers to the time required to execute an algorithm—not to how hard the algorithm might be to understand

**HW**

1. Find the factors of 66,043.

66043=1\*\_\_\_\_\*\_\_\_\_\*\_\_\_\_\*…

211, 313

1. Using the public keys n = 91 and e = 5, encrypt the binary bit string 101. Give your answer in the form of a binary bit string.
   * First 101 is the binary representation for 5 (which is less than 91, as required).
   * Then compute 5e %91= 55 %91=3125%91=31.
   * The encrypted message is 11111, which is the binary representation of 31.
2. Using the private keys n = 91 and d = 29, decrypt the binary bit string 10. Give your answer in the form of a binary bit string.
   * If we received the message 10, we would recognize this as the value 2.
   * Then compute the value 2d % 91= 229 %91= 536870912 %91=32
   * The original message is 100000 (it is 32).
3. Find the appropriate value for the decrypting keys n and d in an RSA public-key cryptography system based on the primes p = 7 and q = 19 and the encryption key e = 5.
   * p= 7 and q=19, so n=7\*19=133.
   * Since e=5, we want 5d= k(7-1)(19-1)+1=k(6)(18)+1=108k+1 for some positive integer k.
   * This implies d=(108k+1)/5. We can see that k=1 or k=2 will not work. But if k=3, then d=(324+1)/5=65.