Zachary J. Bell

Ms. Roy

CS 1401

Challenge Lab 1

**Pseudo Code**

**Algorithm Conversion(number, original, target)**

1. Alphabet <- “0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ”
2. j <- number length
3. for (i = 0; i < number length; ++i)
   1. –j
   2. If( number[j] >= 0 and number[j] <= 9 **OR** num[j] >= ‘A’ and num[j] <= ‘Z’)
      1. If number[j] is a character
         1. baseTen <- (baseTen number[j] – 55 )\* original ^ i
      2. else {baseTen <-( baseTen – 48) \*original^i
   3. else{baseTen <- 0}
4. while (baseTen not 0)
   1. prepend (baseTen % target) to str
   2. baseTen <- baseTen / target
5. return str

**Testing the Code**

Note: I used conditionals to test different extreme values. I tried to use assert statements but javac wasn’t having it. All my conditional tests passed.

System.out.println((Conversion("475762", 10, 2).equals("1110100001001110010")) ? "true" : "((10, 2) == 0) failed");

System.out.println((Conversion("1E", 16, 2).equals("11110")) ? "true": "(16, 2) failed");

System.out.println((Conversion("101010", 2, 3).equals("1120")) ? "true" : "(2, 3) failed");

System.out.println((Conversion("111011110", 2, 12).equals("33A")) ? "true" : "(2, 12) failed");

System.out.println((Conversion("6715317", 8, 14).equals("351419")) ? "true" : "(8, 14) failed");

System.out.println((Conversion("2D4A9E", 15, 13).equals("5BA0CC")) ? "true" : "(15, 13) failed");

System.out.println((Conversion("ZZZZZZ", 15, 13).equals("")) ? "true" : "(ZZZZZZ) failed");

System.out.println((Conversion("12", -1, 2).equals("")) ? "true" : "(-1, 2) failed");

System.out.println((Conversion("12", 3, 37).equals("")) ? "true" : "(3, 37) failed");