Homework 1

1. [20 points] Use mathematical induction to prove:

   For any natural number n, $n^3 - n$ is divisible by 3

   answer:

   n = 1, 1-1 = 0, 0/3 = 0
   (n = 2, 8-2=6)

   assume n = k works,

   then $(k+1)^3 - k - 1 = k^3 + 3k^2 + 3k + 1 - k - 1 = k^3 - 5k = (k^3 - k) + 6k$

   is divisible by 3

2. [40 points] Write algorithm of calculating Factorial by following the requirements:

   a. Write the algorithm using loop (as oppose to recursion)

   b. Explain what "loop invariant" is and prove the algorithm is correct

   c. Calculate the running time of this algorithm (you can assume any constant)

   d. Write algorithm using recursion

   answer:

   a.

   ```
   Factorial(n) {
      fac = 1
      k = 1
      while (k <= n) {
         fac = fac * k
         k=k+1
      }
   ```
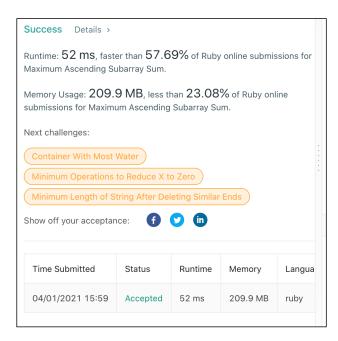
   b. correctness:

   n = 1, it return 1 the algorithm is correct
   loop invariant: each iteration, fac is factorial of (k-1) times k, which is k!
   assuming n =k is correct.
       each iteration, it calculate k! times the new value (k+1), so it calculate a new factorial
       of 1 bigger.

loop will terminate after reach input value n, by this time, fac is factorial of n

c. Factorial(n) {
    if n == 1
        return(1)
    return n*Factorial(n-1)
}

d. running time: $T(n) = T(n-1) + 1$
                  $= T(n-2) + 1 + 1 = \Theta(n)$

3. [40 points] Solve the "Maximum Ascending Subarray Sum" problem (no, 1800) on
   leetcode.com, then follow the requirements:

   a. Make a successful submission. Attach the screen shot image of your submission.

   Note. your submission image should be something like this
         (that is, pass all test cases, instead of an accept page of a particular test case)

   b. Calculate the running time of your algorithm as a function of your input size.
      (Do not just give the result, show how you calculate the result)