HW1 Report – FE 520

Shangxing Jiang

1.1

(1) (2) First replace all special elements (except English letter) : ord() function can convert letter to its ASCII; Because of A~Z: 65-90 and a~z: 97-122, use if statement "if 64<ord(s[i])<91 or 96<ord(s[i])<123" to do it. Then lower() function establishes a new string whose contents are all lowercase. Finally split() function splits the new string.

(3)(4)(5) use boolean judgment -- element in list ? dict[element] += 1 : dict[element] = 1. Then use a for loop to seek the max frequency, however if there are several same biggest frequency, only the first one can be output.

1.2

For Rectangular, just syntax issue. For Square, overwrite parent class's constructor with only one parameter – length. At the same time, length and width both equal length.

2.1

BruteForcely iterate from the initial point, following the rule: ODE(x) > 0, move right; else, move left. I set step = 0.05, iteration limit = 5000, threshold for solution is abs(ODE) < 0.1. Stable point may promise to be found, but not for unstable point.

2.2

Bisection: first we must ensure the signs of ODE(left) and ODE(right) are different, using sign() function. Then let mid= (left + right) / 2. I set threshold abs(ODE(mid)) >0.1 to do while loop.

2.3

The thoughts are as 2.2, except we need multiple solutions. So use a dictionary to store them. The judgement for stability is ODE3(x) < ODE3(x-step) ? Stable : unstable.

Leetcode – P9

First, negative is asymmetric. Then convert int to string, length = 0 or 1 must be symmetric. For general purpose, middlePoint = length/2, use for loop in range(0, middlePoint) to check symmetricality.