[1]Max token class. Given some strings that represent a password, rate the password and get the max score of all the given.

Score: 1 point for letter, 2 point for num, 3 for symbol

1.) “p1” -> 3

2.) “123” -> 6

3.) “p@!” -> 7

[2] decrypt a message. Given an encrypted message, use a function decrypt to get the hidden message.

Mapping:

K 🡪 forward == Zumi.forward(…)

@ 🡪 reverse == Zumi.reverse(…)

b 🡪 Left (left turn)

$ 🡪 Right (right turn )

1 🡪 sound (use speaker to produce a sound)

X 🡪 spin (make Zumi spin in a circle)

print( decrypt("1$KbK") *# -> dList = ["Sound", "Right", "Forward", "Left", "Forward"]*

print( decrypt("@bX") )*# -> dList = [Reverse, left, spin]*

print( decrypt("bK$") ) *# -> dList = [Left, Forward, right]*

[3] Generate own password. The class takes in a length of password and a seed value for a random number generator.

Class functions:

1.) getListOfPasswords

2.) generatePassWords

3.) selectBest

———-

\*\* challenge: set the number of letter, numbers, and special characters

[4] Max point class. Make it apart of the Zumi mission. When Zumi stops, has to solve the puzzle. Apply this to Zumi. starting @ (0,0), drive to the farthest point.

Sorting. Selection sort.

[1] given a file, sort the words in ascii order. min to max.

file0 = { ‘b’, ‘c’, ‘a’}

file1 = { “disaster” , “Meeseeks”, “job”}

file2 = { “i” , “tomorrow”, “will”, “do”, “it”}

[2] given a class names and salaries, sort the employees based on who is paid the most, break ties with alpha name.