Homework (Week 2)

Write two Python functions string\_reverse1(***s***) and string\_reverse2(***s***) that each function takes a character string ***s*** and output its reverse. For example, the reverse of ‘pots&pans’ would be ‘snap&stop’.

Implement the function with two different recursive methods.

Read Class2 slide page 3 and write a few comments separately indicating which lines of your code are the base case(s), which are the recursive calls and why they will make progress towards a base case.

Submit your code, comments, and at least 3 output runs for different sizes of input strings.

Code1：

def string\_reverse1(s):

if len(s) == 0:

return s

else:

return string\_reverse1(s[1:]) + s[0]

s = "Robert"

print ("The original string is : ",end="")

print (s)

print ("The reversed string(using recursion) is : ",end="")

print (string\_reverse1(s))

"""

String is passed as the argument to a recursive function to reverse a string.

The base case is that the length of the string equal to 0, the string is returned.

If the length of the string is not 0, the reverse string function is recursively

called to slice the part of the string except the first character and move the

first character to the endof the sliced string.

"""

Output：

runfile('C:/Users/rober/untitled3.py', wdir='C:/Users/rober')

The original string is : cstu

The reversed string(using recursion) is : utsc

runfile('C:/Users/rober/untitled3.py', wdir='C:/Users/rober')

The original string is : 1601 McCathy Blvd

The reversed string(using recursion) is : dvlB yhtaCcM 1061

runfile('C:/Users/rober/untitled3.py', wdir='C:/Users/rober')

The original string is : Robert

The reversed string(using recursion) is : treboR

Code2

# -\*- coding: utf-8 -\*-

"""

Created on Wed Jan 20 14:11:40 2021

@author: robert

"""

def string\_reverse2(in\_string, rev\_string):

if in\_string=="":

return rev\_string

else:

rev\_string+=in\_string[-1]

print(rev\_string, end=' \_ ')

print(in\_string[:-1])

return string\_reverse2(in\_string[:-1], rev\_string)

in\_string=input("input string is: ")

rev\_string=string\_reverse2(in\_string, '')

print(f"reverse of {in\_string} is {rev\_string}")

""" this program has a recursively function to add and assign the last character form in\_string to

rev\_string until the in\_string equal to "". I add two print statements in the function just to see the intermediate result.

"""Output

runfile('C:/Users/rober/string\_reverse2.py', wdir='C:/Users/rober')

input string is: 123456

6 \_ 12345

65 \_ 1234

654 \_ 123

6543 \_ 12

65432 \_ 1

654321 \_

reverse of 123456 is 654321

runfile('C:/Users/rober/string\_reverse2.py', wdir='C:/Users/rober')

input string is: cstu

u \_ cst

ut \_ cs

uts \_ c

utsc \_

reverse of cstu is utsc

runfile('C:/Users/rober/string\_reverse2.py', wdir='C:/Users/rober')

input string is: 1601 McCathy Blvd

d \_ 1601 McCathy Blv

dv \_ 1601 McCathy Bl

dvl \_ 1601 McCathy B

dvlB \_ 1601 McCathy

dvlB \_ 1601 McCathy

dvlB y \_ 1601 McCath

dvlB yh \_ 1601 McCat

dvlB yht \_ 1601 McCa

dvlB yhta \_ 1601 McC

dvlB yhtaC \_ 1601 Mc

dvlB yhtaCc \_ 1601 M

dvlB yhtaCcM \_ 1601

dvlB yhtaCcM \_ 1601

dvlB yhtaCcM 1 \_ 160

dvlB yhtaCcM 10 \_ 16

dvlB yhtaCcM 106 \_ 1

dvlB yhtaCcM 1061 \_

reverse of 1601 McCathy Blvd is dvlB yhtaCcM 1061