Day 2

Assignment -2

1 .Binary

bin = 0111

print("The decimal value of", dec, "is:")

print(dec(dec), "in decimal.")

print(oct(dec), "in octal.")

print(hex(dec), "in hexadecimal.")

2.Combine and store

test\_list1 = [1, 5, 6, 9, 11]

test\_list2 = [3, 4, 7, 8, 10]

print("The original list 1 is : " + str(test\_list1))

print("The original list 2 is : " + str(test\_list2))

size\_1 = len(test\_list1)

size\_2 = len(test\_list2)

res = []

i, j = 0, 0

while i < size\_1 and j < size\_2:

if test\_list1[i] < test\_list2[j]:

res.append(test\_list1[i])

i += 1

else:

res.append(test\_list2[j])

j += 1

res = res + test\_list1[i:] + test\_list2[j:]

print("The combined sorted list is : " + str(res))

3.Electricity bill

units = int(input("Enter the number of units consumed: "))

if units < 100:

amount = 20

elif units <= 200:

amount = 20 + (units - 100) \* 3

elif units <= 400:

amount = 20 + 300 + (units - 200) \* 5

elif units <= 500:

amount = 20 + 300 + 1000 + (units - 400) \* 7

else:

amount = 20 + 300 + 1000 + 700 + (units - 500) \* 10

print("Your electricity bill is: Rs.", amount)

4.Fact

for i in range(10,0,-1):

print(i)

5.Valid parentheses

def check(myStr):

stack = []

for i in myStr:

if i in open\_list:

stack.append(i)

elif i in close\_list:

pos = close\_list.index(i)

if ((len(stack) > 0) and

(open\_list[pos] == stack[len(stack)-1])):

stack.pop()

else:

return "Unbalanced"

if len(stack) == 0:

return "Balanced"

else:

return "Unbalanced"

string = "{[]{()}}"

print(string,"-", check(string))

string = "[{}{})(]"

print(string,"-", check(string))

string = "((()"

print(string,"-",check(string))

6.Searching intrest position

class Solution(object):

def searchInsert(self, nums, target):

"""

:type nums: List[int]

:type target: int

:rtype: int

"""

try:

return nums.index(target)

except:

for i in range(len(nums)):

print(i)

if nums[i] - target > 0:

return i

else:

print( "hello", len(nums))

return len(nums)

7. Longest common prefix

def longestCommonPrefix(self, strs):

longest\_pre = ""

if not strs: return longest\_pre

shortest\_str = min(strs, key=len)

for i in range(len(shortest\_str)):

if all([x.startswith(shortest\_str[:i+1]) for x in strs]):

longest\_pre = shortest\_str[:i+1]

else:

break

return longest\_pre

8.Find the index of the Frist occurance

string = "This guy is a crazy guy."

print(string.index("guy"));

9.roman into integers

class Solution(object):

def romanToInt(self,s):

roman = {'I':1,'V':5,'X':10,'L':50,'C':100,'D':500,'M':1000,'IV':4,'IX':9,'XL':40,'XC':90,'CD':400,'CM':900}

i = 0

num = 0

while i < len(s):

if i+1<len(s) and s[i:i+2] in roman:

num+=roman[s[i:i+2]]

i+=2

else:

num+=roman[s[i]]

i+=1

return num

ob1 = Solution()

print(ob1.romanToInt("III"))

print(ob1.romanToInt("CDXLIII"))

10.merge two sorted lists

est\_list1 = [1, 5, 6, 9, 11]

test\_list2 = [3, 4, 7, 8, 10]

print("The original list 1 is : " + str(test\_list1))

print("The original list 2 is : " + str(test\_list2))

size\_1 = len(test\_list1)

size\_2 = len(test\_list2)

res = []

i, j = 0, 0

while i < size\_1 and j < size\_2:

if test\_list1[i] < test\_list2[j]:

res.append(test\_list1[i])

i += 1

else:

res.append(test\_list2[j])

j += 1

res = res + test\_list1[i:] + test\_list2[j:]

print("The combined sorted list is : " + str(res))