RB_EE6770_Homework_1

August 20, 2024

EE 6770 Fall 2024: Homework 1

1. Write your answer in the cell provided under each question.
2. **Write comments** in the code to explain your thoughts.
3. **Show your execution result**.
4. **Do your own work.**

0.0.1 Submission:

- **Submit this notebook file and the pdf version** - remember to add your name in the filename.
- Deadline: 11:59 pm, 8/25 (Sunday)

First, check the version of your Python.

[ ]:

```
## Your code here.
import sys
print(sys.version)
```

```
3.11.9 | packaged by Anaconda, Inc. | (main, Apr 19 2024, 16:40:41) [MSC v.1916
64 bit (AMD64)]
```

0.0.2 Q1: Fibonacci Sequence (30 Points)

0.0.3 Q1.1 Write a Python function named `Fibonacci` to generate a Fibonacci sequence of length N.

Recall that the recursive algorithm is:

```
F(0) = 0
F(1) = 1
F(n) = F(n-1) + F(n-2), for n>=2
```

To show your result, assume N = 20

[ ]:

```
## Your code here.
def Fibonacci(N):
    accum = 0
    a = 1
    b = 2
    for i in range(0, N):
        if i == 0:
            accum = 0
        elif i == 1:
            accum += 1
        elif i == 2:
            accum += 2
        else:
            accum = b + a
            a = b
            b = accum
    return accum
print(Fibonacci(20))
```

```
6765
```

0.0.4 Q1.2 Write a function named isFibonacci to test if an input integer N is a Fibonacci number or not.

You can search the web and find out the criteria, or, you can generate Fibonacci numbers until the generated number is greater than or equal to N.

To show your result, try those two numbers: N = 5702887 and N = 74319

[ ]:

```
## Your code here.
def isFibonacci(N):
    if (N == 0):
        return True
    else:
        a1 = 5 * (N**2) + 4
        a2 = 5 * (N**2) - 4
        b1 = int(a1**.5)
        b2 = int(a2**.5)

        if (b1**2 == a1) or (b2**2 == a2):
            return True
        else:
            return False

print(isFibonacci(74319))
print(isFibonacci(5702887))
```

```
True
False
True
```

0.0.5 Q2: (10 Points): With the following list of integers, remove all occurrences of 0 from the list.

```
my_list = [4, 0, 20, 0, 5, 20, -100, 0, 20]

Hint: you can use del(), remove(), pop() for this purpose.
```

[ ]:

```python
### Your code here.

my_list = [4, 0, 20, 0, 5, 20, -100, 0, 20]
for i in my_list:
    if (i == 0):
        my_list.pop(my_list.index(i))
print(my_list)
```

```
[4, 20, 5, 20, -100, 20]
```

0.0.6 Q3: Happy Twosday! (30 Points)

February 22, 2022, was a rare day that all five digits of the date, 22222, are twos. The string is both **palindrome** and **symmetric**. Recall that - A sequence is said to be `palindrome` if one half of the string is the reverse of the other half, e.g., “madam”. - A sequence is said to be `symmetrical` if both halves of the string are the same, e.g., “abcabc”. - If the length of a sequence is an odd number, the middle element in the sequence is not considered.

0.0.7 Q3.1 Write a Python program to confirm that February 22 (22222) is indeed palindrome and symmetrical.

[ ]:

```python
## Your code here.
def Palindrom(N):
    pal = str(N)
    length = len(pal) - 1
    for i in range(0, length):
        a = pal[i]
        b = pal[length - i]
        if (a != b):
            return False
    return True

def Symetric(N):
    sym = str(N)
    offset = int(len(sym) / 2) + (len(sym) % 2)
    span = int((len(sym) / 2))

    for i in range(0, span):
        a = sym[i]
        b = sym[i + offset]
        if (a != b):
            return False
    return True


print(Palindrom(22222))
print(Symetric(22222))
```

```
True
True
```

0.0.8 Q3.2 There are more dates in February 2022 that are also both palindrome and symmetric (such as 22122).

Write a Python loop to find out all the other dates.

[ ]:

```python
## Your code here.
def Palindrom(N):
    pal = str(N)
    length = len(pal) - 1
    for i in range(0, length):
        a = pal[i]
        b = pal[length - i]
        if (a != b):
            return False
    return True

def Symetric(N):
    sym = str(N)
    offset = int(len(sym) / 2) + (len(sym) % 2)
    span = int((len(sym) / 2))

    for i in range(0, span):
        a = sym[i]
        b = sym[i + offset]
        if (a != b):
            return False
    return True

for i in range (20122, 22922):
    if (Palindrom(i)):
        print(i, " is a palindrom")
    if (Symetric(i)):
        print(i, " is symmetric")
```

```
20202  is a palindrom
20220  is symmetric
20302  is a palindrom
20320  is symmetric
20402  is a palindrom
20420  is symmetric
20502  is a palindrom
20520  is symmetric
20602  is a palindrom
20620  is symmetric
20702  is a palindrom
20720  is symmetric
```

```
20802  is a palindrom
20820  is symmetric
20902  is a palindrom
20920  is symmetric
21012  is a palindrom
21021  is symmetric
21112  is a palindrom
21121  is symmetric
21212  is a palindrom
21221  is symmetric
21312  is a palindrom
21321  is symmetric
21412  is a palindrom
21421  is symmetric
21512  is a palindrom
21521  is symmetric
21612  is a palindrom
21621  is symmetric
21712  is a palindrom
21721  is symmetric
21812  is a palindrom
21821  is symmetric
21912  is a palindrom
21921  is symmetric
22022  is a palindrom
22022  is symmetric
22122  is a palindrom
22122  is symmetric
22222  is a palindrom
22222  is symmetric
22322  is a palindrom
22322  is symmetric
22422  is a palindrom
22422  is symmetric
22522  is a palindrom
22522  is symmetric
22622  is a palindrom
22622  is symmetric
22722  is a palindrom
22722  is symmetric
22822  is a palindrom
22822  is symmetric
```

0.0.9 Q4: Python Class Practice (25 Points)

A Python class called `ShoppingList` is created to perform some simple tasks: 1. Add items and prices to the list 2. Calculate pre-tax sub-total 3. Calculate after-tax total

Here is a partially completed `ShoppingList(dict)`, where the input is a dictionary.

[ ]:

```
class ShoppingList(dict):

    tax_rate = 0.06

    def __init__(self):
        self = {}

    def addToList(self, item, price):
        self.update({item: price})    # Add item with its price and number of␣
 ↪items bought to the shopping list

    def Subtotal(self):    # Add your code for pre-tax total
        sum = 0
        for item, price in self.items():
            sum += price[0] * price[1]
        return sum



    def AfterTax_Total(self):   # Add your code for after-tax total
        sum = 0
        for item, price in self.items():
            sum += price[0] * price[1]
        sum += sum * self.tax_rate
        return sum


list = {'Chia Seeds': (2.5, 2), 'avocado': (1.25, 3), 'potato': (3, 3), 'beer':
 ↪(9.99, 1)}
YinList = ShoppingList()
for item, price in list.items():
    YinList.addToList(item, price)

print("Subtotal = $%.2f" % YinList.Subtotal())
print("After Tax = $%.2f" % YinList.AfterTax_Total())
```

```
Subtotal = $27.74
After Tax = $29.40
```

Here is Dr. Yiin's shopping list:

`{'Chia Seeds': (2.5, 2), 'avocado': (1.25, 3), 'potato': (3, 3), 'beer':(9.99, 1)}, where (a,b`

Please complete the two missing methods, `Subtotal` and `AfterTax_Total` and answer the following two questions:

1. Calculate the subtotal and after-tax total of Dr. Yiin's shopping list (show two decimal numbers).

2. The tax rate in Alpharetta is 7.75%, instead of 6%. Please revise the code, however, without modifying the class variable "tax_rate".

[ ]:

```
## Your code here
```