

Python Hw1

Total 50 points, each 5 points

1. Write a python function `dayOfWeek(m,d,y)` that computes the day of week by using the following formula

$$N = d + 2m + [3(m+1)/5] + y + [y/4] - [y/100] + [y/400] + 2$$

Where d is day, m is month, y is year and $N\%7$ will give you the day of the week.

For example, if $d=16$, $m=6$, $y=2014$ then $N\%7$ is 2 so it is Monday. Note first day of the week is Sunday

2. Write a python function `kthDigit(x, k)` that will compute the k th digit of an integer x . For example the 5th digit of 123456 is 1, while 1st digit of 123456 is 5, and 0th digit of 123456 is 6. I.e., count from right to left and the rightmost digit is 0th digit. If value k is greater than or equal to the number of digits of the integer x than return 0. Note parameter k must be a positive integer. If either input parameter is not an integer than return "unknown".
3. Write a function `isEven`, which can check if an integer is even.
4. Write a function call it `isEvenPositiveInt` which takes an integer x and return true if x is positive and also even. Note `isinstance(x, int)` will return True if x is an integer

So

```
>>> isinstance (23, int)
```

```
True
```

```
>>> isinstance (12.34, int)
```

```
False
```

```
>>> isinstance (12.34, float)
```

```
True
```

5. Write a function and call it `isPerfectCube(x)`
Given an integer value x , returns True if it is a perfect cube and False otherwise. That is, return True if there is another integer y such that $x = y^3$. Thus, `isPerfectCube(27)` returns True, but `isPerfectCube(16)` returns False
6. Write a function call it `f(x)`
Given a numerical value x , it will return $x^2 - x$. for example `f(3) = 6`, `f(1) = 0`. Note $^$ means raise to the power of.
7. Write a function call it `g(x)`
Given a numerical value x , it will return $3x^2 - 4.5x$. For example, `g(3) = 13.5`, `g(1.5) = 0`
8. Write a function call it `h(x)`
Given a numerical value x , it will return $\sin(x) + \cos(x)$. For example, `h(0) = 1`, `h(2*pi) - 1.0 < 0.00000001`. The reason is due to approximation of π
9. Write a function call it `bisection(f, low, hi, tol)` which will return a root (zero) of function f . where $f(\text{low})$ and $f(\text{hi})$ have opposite sign. I.e., one is positive and the other negative. The tol is tolerance (epsilon).
10. Write a function call it `harmonicNumber(n)` which will find the sum of harmonic series.
`harmonicNumber(n)` will return $1/1 + 1/2 + 1/3 + \dots + 1/n$.

What to submit:

Make sure to name your homework as *yourLastNameYourFirstNamePY-Hw1.py*

Where *YourFirstName* is your real first name and *yourLastName* is your real last name

Your *yourLastNameYourFirstNamePY-Hw1.py* shall look like the following

```
from math import *
def isEven(x):
    return something

def isEvenPositiveInt(x):
    return something

def dayOfWeek(m,d,y):
    return something

def kthDigit(x, k):
    return something

def isPerfectCube(x):
    return something

def f(x):
    return something

def g(x):
    return something

def h(x):
    return something
def bisection (f, low, hi, tol):
    return something

def harmonicNumber(n):
    return something

# make sure you do more tests
# we reserve the right to test more cases

print("Test hw 1...")
assert (isEven(246810) == True)
assert (isEven(19) == False)
assert (isEvenPositiveInt(231) == False)
assert (isEvenPositiveInt(400) == True)
assert (dayOfWeek(4,1,2020) == 4)
```

```

assert (dayOfWeek(3,26,2015) == 5)
assert (kthDigit (1234,3) == 1)
assert (kthDigit (-1234, 3) == 1)
assert (kthDigit (12, 0) == 2)
assert (kthDigit (123456, 10) == 0)
assert (kthDigit (123456, 8.5) == 'unknown')
assert (kthDigit (123456, -8) == 'unknown')
assert (kthDigit (123.456, 10) == 'unknown')
assert (kthDigit ('hello', 10) == 'unknown')
assert (isPerfectCube(27) == True)
assert (isPerfectCube(16) == False)
assert (isPerfectCube(-729) == True)
assert (f(3) == 6)
assert (f(4.5) == 15.75)
assert (f(-21) == 462)
assert (g(1) == -1.5)
assert (isclose(g(3.2), 16.32, abs_tol=1e-6))
assert (isclose(h(0), 1.0, abs_tol=1e-6))
assert (isclose(bisection (f, 0.1, 10.0, 0.00000001), 1.0, abs_tol=1e-6))
assert (isclose(bisection (g, 0.01, 10.0, 0.00000001), 1.5, abs_tol=1e-6))
assert(isclose(bisection (h, 1.01, 3.02, 0.00000001), (3 * pi/4), abs_tol=0.00000001))
assert(isclose (bisection (h, 6.0, 10.0, 0.00000001), (11 * pi/4), abs_tol=1e-6))
assert(isclose (harmonicNumber (0),0.0, abs_tol=1e-6))
assert(isclose (harmonicNumber (1),1/1.0,abs_tol=1e-6 ))
assert(isclose (harmonicNumber (2),1/1.0 + 1/2.0, abs_tol=1e-6 ))
assert(isclose (harmonicNumber (3),1/1.0 + 1/2.0 + 1/3.0, abs_tol=1e-6))
print ("Pass all tests")

```

Note : `isclose` is a function of `math` module, so you have to import `math` module.

```

>>> isclose (3.14159, pi, abs_tol=1e-5)
True
>>> isclose (3.14159, pi, abs_tol=1e-6)
False
>>> print (pi)
3.141592653589793

```

When to submit
2/6/2020

Where to submit
Blackboard

References

1. Python document
<https://docs.python.org/3/tutorial/controlflow.html#defining-functions>
2. Passing function as argument
<https://stackoverflow.com/questions/1349332/python-passing-a-function-into-another-function>
3. First class object
<https://www.geeksforgeeks.org/first-class-functions-python/>
4. Python handouts on Blackboard