Branch: master ▾

Find file | Copy path

Gitam-Skill-Enhancement-May-2019 / PythonProgramming / PythonNotebooks / 19 May 2019.ipynb

Akash-Sinha 21 May 2019 First Commit

c9bf575 3 days ago

1 contributor

Raw | Blame | History

173 lines (172 sloc) 3.34 KB

Problem Solving and Programming

Day No -

Date -

Day Objectives

1. Objective 1
2. Objective 2
3. Objective 3

Problem 1 : #### Problem Statement 3N + 1 Problem #### Constraints #### Test Cases * Test Case 1 * Test Case 2 * Test Case 3 #### Explanation n for a given n, calculate the length of the series n, n/2 or 3n+1, 4 len(4, 2, 1) = 3 range(i , j+1) i, i + 1, i + 2, ...j

In [7]:

```
import timeit
def cycleLength(n):
    if n == 1:
        return 1
    length = 1
    while(n != 1):
        if n % 2 == 0:
            n /= 2
            length += 1
        else:
            n = 3 * n + 1
            length += 1
    return length

def threeNPlusOne(i, j):
    s = timeit.default_timer()
    '''
    m = 0
    for k in range(i, j + 1):
        cl = cycleLength(k)
        if cl > m:
            m = cl
    '''
    m = max(map(cycleLength, range(i, j+1)))
    print(i, j, m, timeit.default_timer()-s)
    return


threeNPlusOne(900, 1000)
```

```
900 1000 174 0.001459533999877749
```

In [23]:

```
def cycleLength(n):
    length = 1
    while(n != 1):
        if n % 2 == 0:
            n = n / 2
        else:
            n = 3 * n + 1
        length += 1
    return length

def mainSolution(i, j):
    #return max(map(cycleLength, range(i, j+1)))
    lenlist = [cycleLength(k) for k in range(i, j+1)]
    return max(lenlist)

mainSolution(201, 210)
```

Out[23]:

```
89
```

In [ ]: