NATIONAL INSTITUTE OF TECHNOLOGY - KARNATAKA, SURATHKAL

THE INSTITUTION OF ENGINEERS - NITK CHAPTER

ML SMP 2019

Assignment 2 - Python

Write Python programs/functions for the following problems with appropriate names. Add these .py/.ipynb files in a single folder with Your name as the name of the folder. Add this folder to Assignment 2 folder in GitHub.

1. A palindrome is a word or a phrase that reads the same backward as forwards. Make a program that checks if a word is a palindrome. If the word is a palindrome, print 0 to the terminal, -1 otherwise. The word contains lowercase letters a-z and will be at least one character long.

2. A leap year baby is a baby born on Feb 29, which occurs only on a leap year. Define a procedure is_leap_baby that takes 3 inputs: day, month and year and returns True if the date is a leap day (Feb 29 in a valid leap year) and False otherwise. A year that is a multiple of 4 is a leap year unless the year is divisible by 100 but not a multiple of 400 (so, 1900 is not a leap year but 2000 and 2004 are).

3. Binary Search in a list containing only integers.

4. **BONUS:**
   Sudoku [http://en.wikipedia.org/wiki/Sudoku] is a logic puzzle where a game is defined by a partially filled 9 x 9 square of digits where each square contains one of the digits 1,2,3,4,5,6,7,8,9. For this question, we will generalize and simplify the game. Define a procedure, check_sudoku, that takes as input a square list of lists representing an n x n sudoku puzzle solution and returns the boolean True if the input is a valid sudoku square and returns the boolean False otherwise.

   A valid Sudoku square satisfies these two properties:

   1. Each column of the square contains each of the whole numbers from 1 to n exactly once.

   2. Each row of the square contains each of the whole numbers from 1 to n exactly once.

   You may assume the input is square and contains at least one row and column.

   ```
   correct = [[1,2,3],
              [2,3,1],
              [3,1,2]]
   ```

```
incorrect = [[1,2,3,4],
             [2,3,1,3],
             [3,1,2,3],
             [4,4,4,4]]

incorrect2 = [[1,2,3,4],
              [2,3,1,4],
              [4,1,2,3],
              [3,4,1,2]]

incorrect3 = [[1,2,3,4,5],
              [2,3,1,5,6],
              [4,5,2,1,3],
              [3,4,5,2,1],
              [5,6,4,3,2]]

incorrect4 = [['a','b','c'],
              ['b','c','a'],
              ['c','a','b']]

incorrect5 = [ [1, 1.5],
               [1.5, 1]]
```