Lecture 5: Programming in Python

BT 3051 – Data Structures and Algorithms for Biology

Karthik Raman
Department of Biotechnology
Indian Institute of Technology Madras

August 12, 2014

Exercise: Identifying Palindromes

- Check if an input string is palindromic
- Example palindrome strings: noon, racecar, detartrated, Malayalam
- How to check?

palindrome_v1: naïve method

Courtesy: Jennifer Campbell & Paul Gries, University of Toronto

Reverse the string and compare the two strings:

```
def is_palindrome_v1(s):
    ''' (str) --> bool

    Return True if and only if s is a palindrome.
    '''
    return s == reverse(s)

def reverse(s):
    '''(str) --> str

    Return the reverse of the input string s.

    >>> reverse('abcde')
    'edcba'
    '''
    rev_s = list(s)
    rev_s.reverse()
    return ''.join(rev_s)
```

palindrome_v2: Compare first half with second

Courtesy: Jennifer Campbell & Paul Gries, University of Toronto

For strings of odd length, we can ignore the middle character:

```python
def is_palindrome_v2(s):
    ''' (str) --> bool
    Return True if and only if s is a palindrome.

    >>> is_palindrome_v2('detartrated')
    True
    >>> is_palindrome_v2('racer')
    False
    '''
    fh = len(s)//2
    sh = fh + len(s)%2
    return s[:fh] == reverse(s[sh:])

def reverse(s):
    '''(str) --> str
    Return the reverse of the input string s.
    '''
    return ''.join(reversed(s))
```

palindrome_v3: Compare pairs of characters

Courtesy: Jennifer Campbell & Paul Gries, University of Toronto

For strings of odd length, we can ignore the middle character:

```python
def is_palindrome_v3(s):
    ''' (str) --> bool
    Return True if and only if s is a palindrome.

    >>> is_palindrome_v3('detartrated')
    True
    >>> is_palindrome_v3('racer')
    False
    '''
    i = 0
    j = len(s) - 1
    while i < j and s[i] == s[j]:
        i = i + 1
        j = j - 1

    return j <= i
```

What does the docstring tell us?

The *Pythonic* way ...

```python
def is_palindrome_v4(s):
    ''' (str) --> bool

    Return True if and only if s is a palindrome.

    >>> is_palindrome_v4('detartrated')
    True
    >>> is_palindrome_v4('noon')
    False
    >>> is_palindrome_v4('racer')
    False
    '''
    return s == s[len(s)::-1]
    #better still, return s==s[::-1]

import doctest
doctest.testmod(verbose=True)
```

Which version is the best?

Can we now identify palindromic DNA sequences?

- Is GAATTC palindromic?
- 44-bp palindrome found between the genes *cdc53* and *lys21* on chromosome IV in *Saccharomyces cerevisiae*[a]
- Protein sequences are also known to be palindromic

[a] Lisnić B *et al* (2005) Palindrome content of the yeast *Saccharomyces cerevisiae* genome *Curr Genet* **47**:289–97

Self-assessment Exercise

- Given a stretch of DNA (5’->3’), return `True` if and only if it is palindromic
- Outcome:
  - Practice string manipulations in Python
  - Actual biological application
  - Write functions and return values
  - Implement `doctest` for some cases