

Object-oriented Programming – Homework 7

Part A.

Use subversion to create, add, and commit a new directory called “hw7” to your section of the class repository.

1. Use subversion to create and add two new files called “hw7_objects.py” to your section of the class repository.
2. Inside hw7-objects.py, define a new class:
 - a. **Range**, which represents the start and end position (in interbase) of a segment of genomic sequence. For the definition of interbase, see: http://gmod.org/wiki/Introduction_to_Chado#Interbase_Coordinates but for now, don't worry about strand information.
 - b. **Range** should have “setters” and “getters” called: **getStart**, **setStart**, **getEnd**, **setEnd**, **getLength**.
 - c. Add a constructor to **Range** that accepts two arguments: “start” and “end” and uses your “setter” methods to store these values as the start and end of the range, where end >= start and both are non-negative integers.
 - d. Add a new method “overlaps” that accepts another **Range** instances and returns “true” if the two Ranges overlap, according to the following definition:

Two ranges overlap if they have some sequence bases in common. Consider the following example:

Genomic sequence:

```
a t c t g g t c c t g g c c c a a a t t a
| range A | A = Range(1,7)
          | range B | B = Range(6,11)
                    | range C | C = Range(11,16)
```

```
A.overlaps(B) returns True
A.overlaps(C) returns False
A.overlaps(B) returns False
```

Part B.

Use subversion to create, add, and commit a new file to hw7 called hw7.py. Create a single method called testOverlap that accepts two lists with integer start and end values, creates Range objects from them, and returns True if they overlap and False if not. For example,

```
>>> lst = [[1,5],[5,10]]
>>> import hw7 as h
>>> hw7.testOverlap(lst)
False
```