

Essential Computing for Bioinformatics

Lecture 6

High-level Programming with Python

Part III: Files and Directories

Bienvenido Vélez

UPR Mayaguez

Reference: How to Think Like a Computer Scientist: Learning with Python (Ch 11)

Outline

- Text Files
- Reading from Text Files
- Writing to Text Files
- Examples

Text Files

- Persistent (non-volatile) storage of data
- Needed when:
 - data must outlive the execution of your program
 - data does not fit in memory (external algorithms)
 - data is supplied in batch form (non-interactive)
- Files are stored in your hard drive
- Files are maintained by your computer's Operating System (e.g. Linux, Windows, MacOS)

Examples of Text Files

- Word documents
- Html documents retrieved from the web
- XML documents
- FASTA files
- GENBANK file

Text files contain a sequence of numbers that must be decoded using some standard in order to be converted to string form

Examples of encodings: ASCII, LATIN1, EBCDIC, Unicode

Check http://en.wikipedia.org/wiki/Character_encoding for more info

Reading From Text Files

```
in file="<some-file-name>"
infh=open(infile)
line=infh.readline()
while line:
    #do something with the line
    line=infh.readline()
infh.close()
```

Summary of File Operations

Table 12.1. File methods

Method	Action
<code>read([n])</code>	reads at most n bytes; if no n is specified, reads the entire file
<code>readline([n])</code>	reads a line of input, if n is specified reads at most n bytes
<code>readlines()</code>	reads all lines and returns them in a list
<code>xreadlines()</code>	reads all lines but handles them as a <code>XRangeType</code>
<code>write(s)</code>	writes strings s
<code>writelines(l)</code>	writes all strings in list l as lines
<code>close()</code>	closes the file
<code>seek(offset [, mode])</code>	changes to a new file position=start + offset. start is specified by the mode argument: mode=0 (default), start = start of the file, mode=1, start = current file position and mode=2, start = end of the file

Reading from Files II

Structured Text Files

```
4657 GCGTAT
5739 GGGGCCTAA
6123 TTTTACGTACGCGGGCC
...
```

```
def loadSequencesFromFile(filename):
    seq_dict={}
    infh= open(filename)
    for line in infh.xreadlines():
        fields = line.split()
        code = fields[0]
        seq = fields[1]
        seq_dict[code] = seq
    infh.close()
    return seq_dict
```

Writing to Text Files

```
def storeSequenceComplementsFile(sequences, filename):  
    ofh = open(filename, "w")  
    for key in sequences.keys():  
        print >>ofh, key, sequences[key], complement(sequences[key])  
    ofh.close()
```

Exercises

- Write a function to generate a file of proteins corresponding to a file of sequences