Data Formats

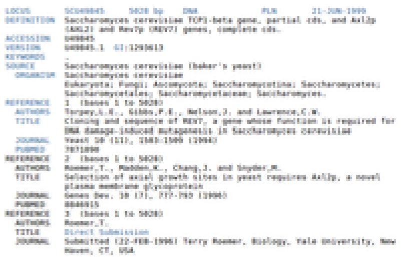
二月 4, 2013, 5:51 a.m. by Rosalind Team

Topics: Bioinformatics Tools, Sequence Analysis

Same Data, Different Formats

A number of different data presentation formats have been used to represent genetic strings. The history of file formats presents its own kind of evolution: some formats have died out, being replaced by more successful ones. Three file formats are currently the most popular:

- FASTA (.fas, .fasta): used by practically all modern software and databases, including ClustalX, Paup, HyPhy, Rdp, and Dambe.
- **NEXUS** (.nex, .nexus, .nxs): used by Paup, MrBayes, FigTree, and SplitsTree.
- **PHYLIP**, or "**Phyl**ogeny **I**nference **P**ackage" (.phy): used by Phylip, Tree-Puzzle, PhyML, and a number of databases.



```
LOCUS       SCU49845     5028 bp    DNA             PLN       21-JUN-1999
DEFINITION  Saccharomyces cerevisiae TCP1-beta gene, partial cds, and Axl2p
            (AXL2) and Rev7p (REV7) genes, complete cds.
ACCESSION   U49845
VERSION     U49845.1  GI:1293613
KEYWORDS    .
SOURCE      Saccharomyces cerevisiae (baker's yeast)
  ORGANISM  Saccharomyces cerevisiae
            Eukaryota; Fungi; Ascomycota; Saccharomycotina; Saccharomycetes;
            Saccharomycetales; Saccharomycetaceae; Saccharomyces.
REFERENCE   1  (bases 1 to 5028)
  AUTHORS   Torpey,L.E., Gibbs,P.E., Nelson,J. and Lawrence,C.W.
  TITLE     Cloning and sequence of REV7, a gene whose function is required for
            DNA damage-induced mutagenesis in Saccharomyces cerevisiae
  JOURNAL   Yeast 10 (11), 1503-1509 (1994)
  PUBMED    7871890
REFERENCE   2  (bases 1 to 5028)
  AUTHORS   Roemer,T., Madden,K., Chang,J. and Snyder,M.
  TITLE     Selection of axial growth sites in yeast requires Axl2p, a novel
            plasma membrane glycoprotein
  JOURNAL   Genes Dev. 10 (7), 777-793 (1996)
  PUBMED    8846915
REFERENCE   3  (bases 1 to 5028)
  AUTHORS   Roemer,T.
  TITLE     Direct Submission
  JOURNAL   Submitted (22-FEB-1996) Terry Roemer, Biology, Yale University, New
            Haven, CT, USA
```

Figure 1. An example of a GenBank record header.

A simple reference on file formats can be found here.

In this problem, we will familarize ourselves with FASTA. We will save the other two formats for later problems.

In FASTA format, a string is introduced by a line that begins with '>', followed by some information labeling the string. Subsequent lines contain the string itself; the next line beginning with '>' indicates that the current string is complete and begins the label of the next string in the file.

GenBank hosts its own file format for storing genome data, containing a large amount of information about each interval of DNA. The GenBank file describes the interval's source, taxonomic position, authors, and features (see **Figure 1**).

A sample GenBank entry can be found here. You may export an entry to a variety of file formats by selecting the appropriate file format under the `Send To:` dropdown menu at the top of the page.

Problem

GenBank can be accessed here. A detailed description of the GenBank format can be found here. A tool, from the SMS 2 package, for converting GenBank to FASTA can be found here.

Given: A collection of n ($n \leq 10$) GenBank entry IDs.

Return: The shortest of the strings associated with the IDs in FASTA format.

Sample Dataset

```
FJ817486 JX069768 JX469983
```

Sample Output

```
>JX469983.1 Zea mays subsp. mays clone UT3343 G2-like transcription factor mRNA,
partial cds
ATGATGTATCATGCGAAGAATTTTTCTGTGCCCTTTGCTCCGCAGAGGGCACAGGATAATGAGCATGCAA
GTAATATTGGAGGTATTGGTGGACCCAACATAAGCAACCCTGCTAATCCTGTAGGAAGTGGGAAACAACG
```

```
GCTACGGTGGACATCGGATCTTCATAATCGCTTTGTGGATGCCATCGCCCAGCTTGGTGGACCAGACAGA
GCTACACCTAAAGGGGTTCTCACTGTGATGGGTGTACCAGGGATCACAATTTATCATGTGAAGAGCCATC
TGCAGAAGTATCGCCTTGCAAAGTATATACCCGACTCTCCTGCTGAAGGTTCCAAGGACGAAAAGAAAGA
TTCGAGTGATTCCCTCTCGAACACGGATTCGGCACCAGGATTGCAAATCAATGAGGCACTAAAGATGCAA
ATGGAGGTTCAGAAGCGACTACATGAGCAACTCGAGGTTCAAAGACAACTGCAACTAAGAATTGAAGCAC
AAGGAAGATACTTGCAGATGATCATTGAGGAGCAACAAAAGCTTGGTGGATCAATTAAGGCTTCTGAGGA
TCAGAAGCTTTCTGATTCACCTCCAAGCTTAGATGACTACCCAGAGAGCATGCAACCTTCTCCCAAGAAA
CCAAGGATAGACGCATTATCACCAGATTCAGAGCGCGATACAACACAACCTGAATTCGAATCCCATTTGA
TCGGTCCGTGGGATCACGGCATTGCATTCCCAGTGGAGGAGTTCAAAGCAGGCCCTGCTATGAGCAAGTC
A
```

Programming Shortcut

Here we can again use the `Bio.Entrez` module introduced in “GenBank Introduction”. To search for particular access IDs, you can use the function `Bio.Entrez.efetch(db, rettype)`, which takes two parameters: the `db` parameter takes the database to search, and the `rettype` parameter takes the data format to be returned. For example, we use "nucleotide" (or "nuccore") as the `db` parameter for Genbank and "fasta" as the `rettype` parameter for FASTA format.

The following code illustrates `efetch()` in action. It obtains plain text records in FASTA format from NCBI's [Nucleotide] database.

```
>>>from Bio import Entrez
>>>Entrez.email = "your_name@your_mail_server.com"
>>>handle = Entrez.efetch(db="nucleotide", id=["FJ817486, JX069768, JX469983"
], rettype="fasta")
>>>records = handle.read()
>>>print records
```

To work with FASTA format, we can use the `Bio.SeqIO` module, which provides an interface to input and output methods for different file formats. One of its main functions is `Bio.SeqIO.parse()`, which takes a handle and format name as parameters and returns entries as SeqRecords.

```
>>>from Bio import Entrez
>>>from Bio import SeqIO
>>>Entrez.email = "your_name@your_mail_server.com"
>>>handle = Entrez.efetch(db="nucleotide", id=["FJ817486, JX069768, JX469983"
], rettype="fasta")
>>>records = list (SeqIO.parse(handle, "fasta")) #we get the list of SeqIO ob
jects in FASTA format
>>>print records[0].id  #first record id
gi|227437129|gb|FJ817486.1|
>>>print len(records[-1].seq)  #length of the last record
771
```

Congratulations You solved this problem (attempt #1). Now you may like to try problems “Pairwise Global Alignment”, “FASTQ format introduction”, “Protein Translation”.