

**Homework 1**  
**Dynamic Programming Alignment**  
**Due: January 29, 2009**  
**50 points**

Your task is to write and test a program in C, C++, Java, Perl, Python or Matlab for Andrew Unix, Linux, MacOs or Windows to perform a dynamic programming alignment for two sequences using a similarity matrix, a gap opening penalty and a gap extension penalty. The program should read required input values input from the command line and write required outputs to a text file (it should not use any interactive or graphical user interface). **Output should be limited to the items requested (and should not include prompts of any type).** The program should:

1. parse from the command line the following inputs
  - a. two Genbank identifiers (gi) for proteins of interest, one per line (you should assume that the gi numbers are for proteins and not for nucleotide sequences)
  - b. the name of a similarity matrix file, which is a text file with 26 lines of 26 comma-separated integer values (in alphabetical order) (an example PAM250.csv is provided)
  - c. a gap opening penalty and a gap extension penalty (integers)
  - d. the name of an output file to be created
2. open the output file for writing all outputs
3. download the peptide sequence for those entries from NCBI in a format of your choice (the Protein database is available at <http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Protein>)
4. read the sequences into internal character strings and output the total number of amino acids in each as two integers
5. read the similarity matrix and output the sum of all entries as an integer on a separate line
6. carry out **global** alignment of the two sequences by dynamic programming **with end gap penalties** (e.g., gaps at the end of one sequence are penalized in the same way as internal gaps)
7. output the maximum alignment score as an integer on a separate line
8. output all alignments that have that score, in three lines per alignment with each line truncated at 255 characters (see details below)
9. output the number of alignments with the maximum score (as an integer on a separate line)

Be sure to use a modular design, follow good programming practice and provide good internal documentation for your code. **Be sure to test your program thoroughly and make sure that the program traps exceptions resulting from invalid or missing inputs.**

Create a single zip, gzip, tar, or jar file containing all of the items below and submit it through the Blackboard drop box. Name the file with your Andrew username and the homework number, e.g., smith-hw1.zip

Provide the following files:

1. The source files (including required shell scripts listed below, include files, etc.)
2. A thorough description of the test procedures you used to verify that the program operates properly, including results from these tests and how that you know the results are correct
3. Any scripts or similarity matrix files that you used in testing

### Summary of output from program

Total number of amino acids in the first sequence (positive integer)

Total number of amino acids of the 2<sup>nd</sup> sequence (positive integer)

Sum of similarity matrix

Maximum Alignment Score (integer)

Sequence Alignments (text) (3 lines for each alignment, all lines truncated at 255 chars)

first line with first sequence, with hyphens (-) for gaps

second line with vertical bars (|) for matches and spaces for mismatches

third line with second sequence, with hyphens (-) for gaps

Number of Maximum Alignments (positive integer)

### About program grading

The homework will be graded using grading scripts so specific naming and input/output conventions must be followed in order to receive credit.

You must include a shell script named “compile” that compiles the program (if compilation is not necessary, make an empty “compile” file) and a shell script named “runme” that takes the arguments from the command line and runs your program using them.

An example output file for the command “runme 46947342 89577 PAM250.csv 1 1 output.txt” is:

#### output.txt

```
10
20
-568
52
MPMILGYWD-----I-----
| | | | | | | | | |
-PMILGYWDIRGLAHAISLLL
MPMILGYWDI-----
| | | | | | | | | |
-PMILGYWDIRGLAHAISLLL
2
```