

CS 5045 – Computation for the Life Sciences
Fall 2003

Exam 02

November 17, 2003

Note: good programs are carefully commented.

1

Write a hash-based perl program that reads a list of words from the STDIN, and reports the words in ASCIIbetical order together with the number of times each word was present in the list.

2

Write a Perl program which creates a dbm database which reads first names (used as keys) followed by family names (used as values) from the console. Write a second Perl program which opens the database and prints each (key,value) pair. Hint: use `dbmopen` and `dbmclose` to open and close the database. The `each` function allows to iterate through all entries in a hash.

3

Write a Perl program to sort an array that contains both numbers and strings. In the sorted array all numbers come first in increasing order, followed by all the strings in ASCIIbetical order. Hint: use regular expressions to distinguish numbers from strings. Use Perl's built-in sort function. For this you need only to supply a mixedsort subroutine which returns -1, 0, or 1 if \$a should come before \$b, same as \$b, or after \$b respectively.

4

Consider a library database written in a Perl dbm file LIB. The keys are book call numbers. The values are strings containing the following entries separated by semicolons: book title, author name, year, number of pages, and publishing house.

Write a Perl program to read this database, and print a table with the first column containing book names (20 characters), the second author names (20 characters), the third year (4 characters), the fourth the no. of pages (5 characters), the fifth the publishing house (20 characters), and the last the call number (10 characters). Hint: You need to split the string, and to use an appropriate format and printf for writing.

5

Write Perl code to parse the FEATURES/source field in the GenBank record attached. This means to read the field and create a hash %source with keys given by the subfield name and values by the subfield value. For example \$source{'organism'} = "male".

6

Consider the following "selection sort" algorithm (applied to an array of n elements)

```
for ($i=0; $i<$n-1; $i++){
    select the minimum element in the subarray @a[$i..$n-1]
        (this element has index $l)
    swap $a[i] and $a[l]
}
```

Implement the full Perl subroutine which sorts an array of *strings* using the the selection sort algorithm above. (Note that in class we implemented this algorithm for arrays of integers).

The number of operations required by this algorithm is in $\Theta(n^2)$. Explain what does this mean, and why this is indeed the complexity of selection sort.