

Homework 5

Due: 25 February 2025

v20250220-1515

The Perl problems here can be done mostly with information from the `perlintro` perl tutorial, but you should also know of the existence of the built-in functions `lc`, `sqrt`, `sort`, and `keys` (some of which are mentioned on the `perlintro` page but which have more documentation on the `perlfunc` page or via `perldoc -f`).

Your scripts should all include both `use strict;` and `use warnings;` as suggested in the tutorial.

Problem 5.1 — practical

Write a perl script that will accept input either from files given on the command line, or from standard input (that is, don't do anything fancy to force either one). Each line of input will have one number on it. Each line of output should have the number that was input and its square, separated by a tab. At the end (after all input numbers have been processed), the program should output the string "Distance: ", followed by the square root of the sum of the squares of all the numbers input.

For instance, if the input were

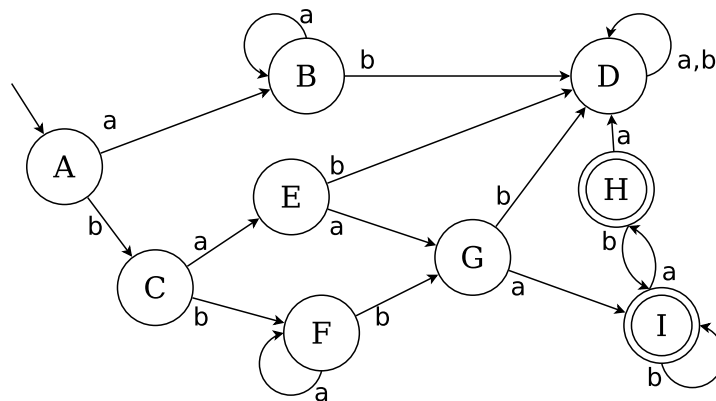
```
3
4
```

the output should be

```
3      9
4      16
Distance: 5
```

Problem 5.2 — theoretical

Consider the following DFA:



Prove that no string in the language accepted by this DFA begins with the symbol a . **Clearly identify** by name the main proof technique used in your proof.

Problem 5.3 — theoretical

Consider the following language:

Strings of length ≥ 2 that start and end with a , and between each pair of a symbols there is either a run of one or more occurrences of the substring bc , or a run of one or more occurrences of the symbol d , or exactly one of the symbol x .

So $abc b c b c a$ and $a d d d d d a$ and $a b c a d a d d a x a b c b c a$ are all in the language; but for instance $a a d a$ and $a b c b a$ and $a x x a$ and a are not.

Construct an NFA that accepts exactly that language, and a DFA that accepts exactly that language, and briefly explain why the NFA can be simpler and/or easier to read than the DFA.

Hand in the file(s) containing the proof and the Perl scripts using the handin script:

```
handin cmsc208 hwk5 proofs.tex myfile.pl otherfile.pl
```

If you want to try putting 5.3 in electronic form too I'll accept it that way, but I think it'll be easier to do that one on paper.

Collaboration policy: **For Problems 5.1:** collaborative. You each have to hand in your own version of the assignment, but you can talk to other people about the problems. Mention in a comment or readme who you worked with. (Still no copying, though.) **For Problem 5.2–5.3:** group work! If you work with other people on this homework, you can just hand in one copy and put all your names on top. There will be a revision cycle for this.