

Homework 11

Due: 22 April 2025

Problem 11.1 — practical

Write the following Haskell functions, following the style we used in class.

- `allGreaterThan n lst` builds a list with all the elements of `lst` that are greater than `n`. (Write this function without using any builtin function other than `:` and `>`.)
- `allLessThan n lst` builds a list with all the elements of `lst` that are less than `n`. (Write this function using the builtin `filter` function.)
- `powersOf n` builds an infinite list with all the natural powers of `n`. (Use explicit case/recursion as we demoed in class.)

As before, include test cases (examples and expected results) in comments.

Problem 11.2 — theoretical

Draw a transition diagram for a Turing machine (with $\Sigma = \{0,1\}$) that accepts strings that have at least two more 1 symbols than 0 symbols. To the extent that states or groups of states in the diagram have meaningful interpretations, label them. Clearly indicate what the tape alphabet Γ is for your TM (and, if not obvious, what each symbol is used for).

Hand in the files containing the Haskell code using the `handin` script:

```
handin cmsc208 hwk11 myfile.hs
```

If you want to put the Turing machine diagram in electronic form too I'll accept them that way, but I think they'll be mostly easier to do on paper. (Please write neatly!)

Collaboration policy: **For Problem 11.1:** 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 Problems 11.2:** 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.

This document was written and prepared without the use of generative AI.