# Assignment 5

This assignment consists of two questions worth 10 points each. These are available on the quiz website `cpsc313.cpsc.ucalgary.ca` in a new category called tasks. You can submit and resubmit your solution as much as you want, and there is not time limit aside from the deadline. The last one you submit will be graded. If you submit something that gets zero marks, you will be warned that this happened.

These require you to submit a Turing machine using the syntax as follows:
STATE,SYMBOL -> STATE,SYMBOL,DIRECTION

use underscore (_) for a blank. Use l and r for direction (left / right). An example is the following:

```
S,_ -> FE,_,r
FE,_ -> ha,_,l
FE,a -> FE,_,r
FE,b -> FE,_,r
```

which erases the tape. You may also use * for a wild card: FE,* -> FE,_,r will replace whatever it sees with a blank, and FE,* -> FE,*,r will write the same symbol as it sees (i.e., doesn't change the contents of the current cell). The rules are processed in order from top to bottom, so the first rule in the list that matches the state-symbol pair (e.g., wildcard) will be used even if a later rule is more refined for the symbol.

You may use the online tool at `http://pages.cpsc.ucalgary.ca/~joel.reardon/313/tools/tm/tm.html` to both test your design and ensure your syntax is correct. *In fact you are encouraged to do this and only submit it after you get it working.*

Also you are encouraged to keep for your records any Turing machines that you design. You may find them useful as reusable components for other quizes / assigments / exams etc.

They were marked by counting correct test cases. One mistake is 7/10, two is 4/10, three is 1/10. Four or more resulting in a zero as well as a warning when you were submitting.

**task 1** It was tested with the following for accept:

```
bbbbaaaaa
aaaaabbbb
a
aba
```

And the following for reject:

```
b
ba
bbbbabbb
aaaabbbb
```

**task 2** It was tested with the following:

```
word -> resulting tape
ab -> _a_b
bba -> _b_b_a
ba -> _b_a
bb -> _b_b
aa -> _a_a
aba -> _a_b_a
a -> _a
epsilon -> _
```