

Bonus Assignment

Due: Friday, November 18, 2016 at 4:00 p.m.

For this bonus assignment, you must construct a Turing machine that takes a lot of steps and then halts.

We shall restrict attention to Turing machines that use the tape alphabet $\Gamma = \{0, 1, X, \sqcup, \triangleright\}$. Your goal is to design a Turing machine which, on input 11111100000, takes as many steps as possible before halting. (Note that 11111100000 is the binary representation of the year 2016.) Your machine must have exactly six states, including the initial state, the accepting state and the rejecting state.

This bonus question is worth up to 4 percent to be added to your final grade. Students who submit a correct Turing machine will get between 1 and 4 bonus points. If two students submit machines that take the same number of steps, each student will get 1 point. Excluding these students, the higher the number of steps your machine takes, the more bonus points you will earn. To earn all 4 points, you will have to make your machine take a very large number of steps.

If your machine takes more than 1,000,000,000 steps on the input string 11111100000, I may not be able to test it by running it, so you will have to provide, in writing, a proof that it does indeed halt (and that it takes lots of steps) in order to receive your bonus points.

Write your Turing machine in a YUTMFF text file named `bonus.txt` and submit it online at <https://webapp.eecs.yorku.ca/submit/> or by using the following command.

```
submit 2001 bonus bonus.txt
```

If your submission is syntactically incorrect or does not follow the specifications of this question (for example if it has too many states) or runs forever on the empty input string, you will receive 0 points.

For this bonus question, you may consult outside sources. A good place to start is the Computer Recreations column by A. K. Dewdney in the August, 1984 issue of *Scientific American* (pp. 19–23). (Note that the Turing machines discussed there use two-way infinite tapes, so you cannot use them directly to answer this question, but you might get some ideas about how to design your own busy beaver.)