Introduction to `flex`

Announcements

- **Written Assignment 1 Out**
  - Due Wednesday, July 6 at 5:00 PM.
  - Covers lexical analysis.
  - Submit in filing cabinet in Gates (through the "Stanford Venture Fund Laboratories" door) or via email.
- Piazzza (online Q&A)
- Office hours will be posted after class.

What is `flex`?

- Automated tool for generating scanners.
- Uses maximal-munch/precedence system described yesterday.
- Internally, builds a DFA from regular expressions.
- Plus several more features...

A Simple **flex** File

```
%%
[A-Za-z]*       printf("Word\n");
[0-9]*          printf("Number\n");
[ \t\n]         ;
.               printf("Symbol\n");
```

Building an Interpreter

- Language: **HQ9+**
- Programs are strings of H, Q, 9, and +:
  - **H** prints “Hello, World!”
  - **Q** prints the program's source code (**Q**uine).
  - **9** prints the song “99 Bottles of Beer on the Wall.”
  - **+** increments the accumulator (which can't be accessed)

Just for Fun

- The Improved Language: **HQ9+LT**
- Programs are strings of H, Q, 9, +, L, and T:
  - **H** prints “Hello, World!”
  - **Q** prints the program's source code (**Q**uine).
  - **9** prints the song “99 Bottles of Beer on the Wall.”
  - **+** increments the accumulator (which can't be accessed)
  - **L** puts the program into an infinite loop.
  - **T** solves the halting problem on the program... by seeing if it contains **L**.