CS 57 Notes

09/12/2018

Introduction to compilers (broad overview).

Compilers consist of: - frontend - lexical analysis - syntactic analysis - outputs IR - backend - takes IR - outputs optimized machine code

Links

- introduction to compilers

09/14/2018

Introduction to git and Makefiles

Git is a distributed version control system, and will be used to submit assignments for the class.

Makefiles are a build system that automatically resolve dependencies in build rules in order to create what you want.

Links

- atlassian git tutorials
- official git reference
- Wooster makefile tutorial
- Make and C tutorial

09/17/2018

Finite automata and regular expressions

Links

- Chalmers intro to FA
- Rochester | Finite Automata
- (Video) Neso Academy | Finite Automata

- Stanford intro to FA
- Good explanation of Epsilon transitions

09/19/2018

Introduction to lex/flex

If you have a Mac:

```
brew install flex
```

To get this tool to show up in your path:

```
echo 'export PATH="/usr/local/opt/flex/bin:$PATH"' >> ~/.bashrc
```

Note that to compile files using flex, you will need to provide the directory to the library and instruct the compiler to use it. So if we had some file `example.l`, we would perform the following steps to compile it with flex:

Note that from this point forward, lex and flex will be use interchangeably.

```
flex example.l  # produces yy.lex.c
clang yy.lex.c -L/usr/local/opt/flex/lib -lfl  # produces a.out
```

Links

- video tutorial on lex/flex
- short lex tutorial
- long, comprehensive tutorial on lex/yacc

09/24/2018

Introduction to bison/yacc

To install bison on MacOS, just install with Homebrew:

```
brew install bison
```

Now, put the path to bison in your $PATH:

```
echo 'export PATH="/usr/local/opt/bison/bin:$PATH"' >> ~/.bashrc
```

Consume grammar files with bison with the following command (assuming we have some `example.y`):

```
bison -d example.y
```

Links

- tutorialspoint intro to everything compilers
- UCR bison tutorial with calculator example