

Tips on how to complete PR1b successfully for CSCE 531 students, Spring 2020

1. Use the latest (August 27, 2019) version of bnfc, version (version 2.8.3), found at <https://github.com/BNFC/bnfc/releases/tag/v2.8.3>. This site is linked to <http://bnfc.digitalgrammars.com/download/>, which includes other, older, versions. In turn, this site is linked to Arne Ranta's textbook website, <http://www.grammaticalframework.org/ipl-book/>. We refer to this textbook as [R].

Haskell

Make sure to read the errata linked to <http://www.grammaticalframework.org/ipl-book/>. An error in the last line of Calculator.hs is described there.

If you install the Haskell Platform (<https://www.haskell.org/platform/>), the Haskell compiler ghc, the lexical analyzer generator Alex, and the parser generator Happy will be installed.

Haskell uses implicit grouping by indentation. The book uses indentation properly. If you do not indent, you need to use braces.

The single quotes around div in program Interpreter.hs (middle of page 25 [R]) are single backquotes: `div` is correct, 'div' is incorrect. (In Haskell, backquotes around a function turn it into an infix operator.)

Windows users need to be aware that they need to use Linux-style echo and make.

echo in Windows PowerShell (which is started by Windows-X) works the intended way; it does not work in the intended way in a Command Prompt terminal; I suspect that the same is true of pipe (|). So: run your Calculator as in the middle of page 26 [R] from PowerShell.

make can be installed as described in <https://stackoverflow.com/questions/2532234/how-to-run-a-makefile-in-windows> (entry of Jul 15 '19 at 15:05):

Step 1: Install the chocolatey package manager for WINDOWS (compatible to Windows 7+ / Windows Server 2003+) here (<https://chocolatey.org/install>)

Step 2: run choco install make

Java

Many thanks to Mr. Justin Baum for allowing the tips below to be shared.

Dear Dr. Valtorta,

After a lot of time tinkering with JLex and CUP, I have it working on Debian on ARM and x86, and also Mac OS. It's very finicky, but its doable.

1. Make a directory in your home called `.java` (`mkdir ~/.java`).

2. Decompress the tar.gz from <http://www2.cs.tum.edu/projects/cup/> . Decompress it in .java. After it should look like such. (Remember to rename the directory as such below)

```
~/java/java-cup/  
├─ java-cup-11b-runtime.jar  
└─ java-cup-11b.jar
```

0 directories, 2 files

3. Run this command

```
cd ~/java && mkdir JLex && cd JLex &&  
(curl https://www.cs.princeton.edu/~appel/modern/java/JLex/current/Main.java > Main.java) && javac  
Main.java && cd ~
```

4. Add this line to ~/.bashrc, (weirdly Java's path needs full paths for .jar's, and can find .class's on its own).

```
export CLASSPATH=./:~/java/java-cup/java-cup-11b.jar:~/java/java-cup/java-cup/java-cup-11b-  
runtime.jar:~/java
```

5. Run `source ~/.bashrc`

6. Test BNFC. I've attached it working on OS X and Debian with script files.

Hopefully this helps those who couldn't get past those issues.

Justin