

CS119 – Activity 1

Run the NoMachine application on your laptop or on one of the lab computers and connect to phoenix. Once you are logged into phoenix, open your home directory that you should see on your desktop. In any directory you can open a terminal by doing a right click and selecting the Open Terminal option. Copy the lab1 folder from my account by using the command:

```
cp -r ~/jillz/cs119/lab1 .
```

Note: The period at the end of the command indicates that the destination for the copy command is the current directory.

You should see the lab1 folder in your directory. Click on the folder to open it and then open up the file Example1.hs in the gedit editor. This file is the only file you need to look at in the directory. Keep the mystery alive and employ abstraction principles by not looking at any of the other code in the folder. (I know, now you want to look at them but don't do it!)

Open a terminal in the lab1 folder. An alternative is to use your previous terminal window and change the directory by using the command: `cd lab1`

In your terminal we want to run the Haskell compiler with the command `ghci` and then when you get the `ghci` prompt use the command `:load Example1.hs` to compile this code.

Now try some of the examples in the notes. Test out the `addS` and `thirdPerson` functions to verify that they work as expected. Make sure that you understand how all the functions in the Words module behave.

Try out the quilt examples in the notes and make sure that you understand how these functions from the Quilts module behave.

Note: When you close a quilt window, the actual drawing does not disappear but it will give you a new prompt within the terminal window for you to type another expression.

Now we will type a new function into Example1.hs. Each time you add a function or make a change, you will need to perform the load command in `ghci` to compile the changes. Complete the following function which deletes the second word from a sentence `s`. Then reload and try out your function in the terminal.

```
delete2 :: Language -> Language
delete2 s = (firstItem s) +++ (_____)
```

Hint: Consider the `butFirst` function.

An example of how to use the function in the terminal:

```
> delete2 (sent "this is a test")
[this a test]
```

Complete the function `min3` which is given three words (all of different lengths) and returns the word with the smallest length. Reload and try out your function in the terminal.

```
min3 :: Language -> Language -> Language -> Language
min3 w1 w2 w3 = if (count w1) < (count w2) then
  if _____
  then w1
  else _____
else if _____
  then _____
  else _____
```

An example of how to use the function in the terminal:

```
> min3 (word "lions") (word "tigers") (word "cats")
cats
```