NOTES ON HASKELL

VAIBHAV KARVE

These notes were last updated July 19, 2018. They are notes taken from my reading of Learn You a Haskell for Great Good! by *Miran Lipovača*.

1. Starting out

(1) All arithmetic operations of `+, -` and `*` work as usual.
(2) Division results in a floating point answer. Integer division can be done by `div`.

```
ghci> 5 / 2
2.5

ghci> div 10 4
2
```

(3) Put parenthesis around negative numbers.

```
ghci> 5 * -3
[ErrorMessage]

ghci> 5 * (-3)
-15
```

(4) `&&`, `||` and `not` represent AND, OR or and NOT operators.
(5) Only compare variables of the same type.

```
ghci> 5 == 4
False

ghci> 5 /= 4
True

ghci> 5 == "4"
[ErrorMessage]

ghci> 5 /= "4"
[ErrorMessage]

ghci> 5 == 5.0
True

ghci> 5 == 5.
[ErrorMessage]
```

(6) *Infix functions* like `+` and `*` take arguments on both sides. Functions that are not infix are *prefix functions*. Examples:

```
ghci> succ 8
9

ghci> min 8 10
8
```

(7) Function arguments are separated by space. Space takes precedence over all other operations.

```
ghci> succ 9 * 10
100

ghci> succ(9 * 10)
91

ghci> succ 9 + max 5 4 + 1 == (succ 9) + (max 5 4) + 1
True
```

(8) Prefix functions can be turned into infix functions by using backticks to write `` `div` ``:

```
ghci> div 10 4
2

ghci> 10 `div` 4
2
```