Written problems

In each of these, simplify the expressions step-by-step to get the final value. If there is an error, say exactly what the problem is. You should show enough detail to make it clear that you know what is going on. In all cases, you should be able to check your answer by typing the expressions into GHCi.

Assume that the following definition has been loaded into GHCi: \( f \ x = 2 \times (x + 1) \)

Now evaluate:

1. \((4 <= 2) \&\& (True || (8 ^ 2 == 3))\)
2. \(8 \times (\text{succ} \ 3) + (\text{max} \ 40 \ 15)\)
3. \(5 + (\text{succ} \ 2 \ 4)\)
4. \"I have " ++ (f 2) ++ " dogs"\)
5. \(f (\text{max} (f 3)) \ 8\)

Programming problems

- Create a function called fToC which converts a temperature in Fahrenheit to Celsius. Do this by subtracting 32, then multiplying by 5/9.
- Create a function called cToF which converts Celsius to Fahrenheit. Do this by multiplying by 9/5, then adding 32.
- Create a function called banger which takes a string and returns the same string, but with an exclamation mark stuck on the beginning and the end.
- Create a function called saywhat which takes a string and repeats it twice. So saywhat "hi mom" is "hi momhi mom"
- Here’s another “built-in” Haskell function: head takes a string and gives the first character (if the string is empty you get an error). tail takes a string and throws away the first character and gives you the rest of the string. Play around with these in GHCi until they make sense to you.
- Create a function called stutter which takes a string and repeats the first letter three times, followed by the rest of the string. So stutter "hi mom" is "hhi mom"
- Create a function called ekzer which takes a string and inserts an “x” after the first character. So ekzer "robert" is "rxobert"