# Worksheet 3 Functional programming Answers

**Task 1**

1. Specify the sets containing all possible output values of these functions:

(a) ascii (x) which takes a character and returns its representation in the ASCII code.

(b) A function which can decide if an integer x is odd or even.

(c) x mod 5 where x is an integer.

(a) {0…255}

(b) {True | False} – the Boolean space

(c) {0,1,2,3,4}

2. What are the widest possible **domains** for the following functions?

(a) Unicode (x) which returns up to 4 bytes

(b) f(x) = 2\*x

(c) hex (x) which returns a single hexadecimal digit

(a) any character in any language

(b) any number, or the set of real numbers ℝ

(c) {0..15}

*Note the question specifies* ***domain****, not* ***co-domain****.*

3. Set up and try out Haskell.

(a) In **interactive mode**, try typing: let greeting = "Hello World", followed by Enter and then greeting [Enter]

(Note that in interactive mode, you must preface any function definition with the word “let”)

(b) Type: let doubleNum x = 2 \* x followed by Enter,

and then doubleNum 8 [Enter]

(c) Use a text editor such as Notepad++ to write a function to find the value of   
quadratic where:

**quadratic = x2 + 5x + 1**

Save your function as quadratic.hs.

(Hint: in Notepad++ you can set the Language to Haskell, which will set the file extension by default.)

(d) Load and run your program to find the value of quadratic when x = 3. What answer does Haskell give?

Program: quadratic x = x \* x + 5 \* x + 1 (or x = x\*\*2 + 5 \* x + 1)

Answer: 25

4. The function returns the area of a circle with radius *r*. Write and test code to define this function. (Haskell has a built-in constant called pi.)

How can you find out, using your function, what value Haskell assigns to pi?

circleArea r = pi \* r \* r

circleArea 1 should return 3.141592653589793

5. Write and test code for a function which calculates the difference in area between 2 circles of radius r1 and r2. (*hint: it might make sense to do this in two steps.)*

Test the function with r1 = 5 and r2 = 3

(see file circleArea.hs)

circleArea r = pi \* r \* r

areaDifference r1 r2 = circleArea r1 – circleArea r2

Note that you don’t have to use the letter r, any variable name will do!

To test the code, load the program and type, for example

areaDifference 5 3

wh ich will return 50.26548245743669

**Task 2**

6. Write a type declaration and definition for these functions, assuming x, y, a and b are type Float. (To test these out, type them into a text editor, then load and run them.)

1. squArea ( x ) which returns the area of a square with sides of x
2. g(x, y) = 2x + 3y
3. sumDiff (a, b) which returns the values of the sum of p and q, and their difference. *Hint: a pair of values, known as a tuple, can be defined as (Int, Int) or   
   (Float, Float) etc.*

squArea :: Float 🡪 Float

squArea x = x \* x

g :: Float 🡪 Float 🡪 Float

g x y = 2 \* x + 3 \* y

sumDiff :: Float 🡪 Float 🡪 (Float, Float)

sumDiff a b = (a + b, a - b)

7. In a text editor, write two functions, to convert from Fahrenheit to Celsius - and back. Include type declarations.

A useful formula is . Test that 20C returns 68F and vice versa.

See file temperatureConvert.hs

f2c :: Float 🡪 Float

f2c f = (f – 32) \* 5 / 9

--

c2f :: Float 🡪 Float

c2f c = (9 \* c) / 5 + 32

8. In interactive mode, type the function f x = 2 \* x with no type declaration.   
(Remember to start with the word “let”.)

Run the functions for these values for x and write down the results:

1. 5
2. 5.0
3. 5.123456789
4. “g”

Note that Haskell returns appropriate number types but gives an error message   
for the character

[1 of 1] Compiling Main ( WS3Q8.hs, interpreted )

Ok, modules loaded: Main.

\*Main> f 5

10

\*Main> f 5.0

10.0

\*Main> f 5.123456789

10.246913578

\*Main> f ("g")

<interactive>:73:1: error:

• No instance for (Num [Char]) arising from a use of ‘f’

• In the expression: f ("g")

In an equation for ‘it’: it = f ("g")

9. The volume of a cylinder with radius r height h is given by the formula

Write a type declaration and definition for a function volCylinder to calculate the volume of a cylinder, and test it with r = 1 and h = 10.

Verify the function type by entering:

:type volCylinder

volCylinder:: Float 🡪 Float 🡪 Float

volCylinder r h = pi \* r \* r \* h -- this is the standard formula

Testing:

**\*Main>** volCylinder 1 10

31.415928

***(See file volCylinder.hs)***

**Extension task**

Spend ten minutes working through <https://wiki.haskell.org/Learn_Haskell_in_10_minutes>

Further reading: see

<http://learnyouahaskell.com/>

<https://wiki.haskell.org/Haskell_in_5_steps>