CS_191/CS_221 Functional Programming I

Coursework 2
Due date: Monday, 20 April 2009

Simple image processing

In this coursework we practice programming with lists, algebraic data types and higher-order functions using the example of some simple image processing.

The questions refer to the definitions given in the lecture slides. All necessary auxiliary programs are in the file SimpleImageProcessing.hs which is available from the course web page. Download this file and begin your coursework with the line

import SimpleImageProcessing

Example runs of the programs need to be shown to the lab supervisors. Please don’t include them in your submission.

Submission: Please submit a printout, signed by both lab partners, in the wooden box on the 2nd floor.

Question 1. Define the following operations on pictures:

(a) appvPic: appending two pictures vertically (second below first).
(b) apphPic: appending two pictures horizontally (side by side).
(c) udPic: flipping a picture upside down.
(d) lrPic: flipping left and right of a picture.
(e) invPic: inverting a picture (complement, that is, pointwise negation).
(f) unionPic: Combining two pictures disjunctively (union, pointwise (||)).
(g) interPic: Combining two pictures conjunctively (intersection, pointwise (&&)).
(h) exclPic: Combining two pictures exclusively (pointwise (/=)).

In questions (a), (b), (f), (g) and (h) it may be assumed that the given pictures are of the same size (but this is not relevant for the implementation).

Hint: All these operations can be defined in one line, possibly using map and zipWith.

[40 marks]

Question 2. Define the following operations on bitmaps:

(a) scaleXBm: Scaling a picture in x-direction by a given factor.
(b) scaleYBm: Scaling a picture in y-direction by a given factor.
(c) udBm: Flipping a bitmap upside down, using (b).
(d) lrBm: Flipping left and right of a bitmap, using (a).

[20 marks]
Question 3. Compute the following data of a shape:

(a) Number of corners.
(b) Perimeter (circumference).
(c) Area.
(d) Diameter (the largest possible distance of two points in the shape).

[40 marks for CS-191 students, 20 marks for CS-221 students]

Question 4 (assessed for CS-221 students only). The logistic map with parameter $r \in [0,4]$

$$\varphi_r : [0,1] \to [0,1], \quad \varphi_r(x) = r \cdot x \cdot (1 - x)$$

$([a,b] = \{x \in \mathbb{R} \mid a \leq x \leq b\})$ is a simple example of a dynamical system with chaotic behaviour. It is used as a demographic model and as a testing function for implementations of real number operations. For more information on the logistic map, see, for example,

http://en.wikipedia.org/wiki/Logistic_map

The chaotic behaviour of the logistic map is due to the fact that for many parameter values $r$ close to 4 and many starting points $x \in [0,1]$ the orbit of the logistic map,

$$\{\varphi^n_r(x) \mid n = 1, 2, 3, \ldots \} = \{\varphi_r(x), \varphi_r(\varphi_r(x)), \varphi_r(\varphi_r(\varphi_r(x))), \ldots \}$$

is a highly irregular set the elements of which (for large $n$) are difficult to compute because of the accumulation of rounding errors.

In the following we only consider the parameter $r = 4$ and set $\varphi := \varphi_4$.

Take pictures of the iterations, $\varphi^n : [0,1] \to [0,1]$, of the logistic map for $n = 1, \ldots, 10$, that is, take pictures of the regions

$$\{(x, y) \mid x, y \in [0,1], y < \varphi^n(x)\} \quad (n = 1, \ldots, 10)$$

[10 marks]

Question 5 (assessed for CS-221 students only). The Mandelbrot set is the set of all complex numbers $c \in \mathbb{C}$ with the property that the mapping

$$f_c : \mathbb{C} \to \mathbb{C}, \quad f_c(z) = z^2 + c$$

has a bounded orbit at 0, that is, the set

$$\{|f^n_c(0)| \mid n = 1, 2, 3, \ldots \} = \{|f_c(0)|, |f_c(f_c(0))|, |f_c(f_c(f_c(0)))|, \ldots \} = \{|c|, |c^2 + c|, |(c^2 + c)^2 + c|, \ldots \}$$

is bounded. The Mandelbrot set is a fractal, that is, a set with an infinitely complex microstructure. For more information, see

http://en.wikipedia.org/wiki/Mandelbrot_set

By identifying a complex number $c = x + i \cdot y \in \mathbb{C}$ with the point $(x, y)$ we may view the Mandelbrot set as a region in the 2-dimensional plane.

Using the setting $((-2, -1.3), (1, 1.3), 160, 80)$ (maybe with a different resolution) take a picture of the following region that approximates the Mandelbrot set:

$$\{c \in \mathbb{C} \mid |f^n_c(0)| < 2 \text{ for all } n \in \{0, 1, 2, \ldots, 20\}\}$$

Reminder on how to calculate with complex numbers:

$$(x + y \cdot i) + (x' + y' \cdot i) = \quad (x + x') + (y + y') \cdot i$$
$$(x + y \cdot i) \cdot (x' + y' \cdot i) = \quad (x \cdot x' - y \cdot y') + (x \cdot y' + y \cdot x') \cdot i$$
$$|x + y \cdot i| = \quad \sqrt{x^2 + y^2}$$

Hence $|x + y \cdot i| < 2$ if and only if $x^2 + y^2 < 4$.

[10 marks]