CPSC 429/529: Functional Programming
Carsten Schürmann
Date: Wed 28 March 2001

Homework 4

due Wed 11 April 2001 at 9:30am

Problem 1 [40 points]

In this assignment, you will implement a simple monadic combinator parser for the tiny lazy functional language from assignment 2. The following table gives a summary of the language syntax.

Feature	Abstract Syntax	Concrete Syntax	Comments
functions	$\lambda x. P$	`\x -> P`	Haskell notation
	$P_1\ P_2$	`app P1 P2`	Juxtaposition
	x	`x`	Lowercase names
recursion	$\mu x. P$	`mu x -> P`	Recursion
integers	$\overline{n}$	`n`	Use `nat` from class
	$P_1 + P_2$	`plus P1 P2`	Haskell Syntax
	ifzero P_0 **then** P_1 **else** P_2	`ifzero P0 then P1 else P2`	
lists	$[\,]$	`[]`	Haskell Syntax
	$P_1 : P_2$	`cons P1 P2`	Haskell Syntax
	head P	`head P`	
	tail P	`tail P`	
	case P_0 **of** $[\,] \rightarrow P_1$ **or** $h : t \rightarrow P_2$	`case P0 of [] => P1 | h : t => P2`	

Hint: Reuse the datatypes you have developed for homework 2, and the monadic parser combinators we have developed in class.

Problem 2 [40 points]

Create a simple arcade game. Create a variety of shapes that drift across the screen from both the left and the right, and use mouse clicks to make them disappear.

Problem 3 [20points]

Write an interim report about the progress on your project, and an outlook on what you want to have accomplished by the end of this semester. In the last week of the semester you are expected to compose a project summary in form of a webpage (to be made public on the course homepage), and give a short presentation in class.