CS 337 **Test 2** **4/4/07**

Open book and notes.

Max points = 75 Time = 75 min Do all questions.

1. (Finite State Machine; 15 points)

   (a) A turnstile is either *locked* or *unlocked*. When the turnstile is locked, a customer can drop a coin into its slot. This causes the turnstile to show a green signal and become unlocked. If the turnstile is unlocked, then a customer can pass through it; then the turnstile becomes locked and a red signal is displayed. Assume that the turnstile begins in the locked state.

   Draw a finite state transducer to specify the behavior of the turnstile. Identify the states, input alphabet and the output alphabet clearly.

   (b) The previous description gives the turnstile behavior in normal cases. Suppose that a customer passes the turnstile when it is locked without dropping a coin. Then the turnstile becomes *superlocked* and an alarm starts sounding. The only way out of this state is (1) the customer drops a coin which causes the turnstile to become locked and a red signal to be displayed, or (2) a technician presses a reset button which causes the turnstile to become locked.

   Redraw the machine.

2. (Finite State Machine; 15 points) The machine in Figure 1 accepts only and all binary strings in which any occurrence of 0 is immediately followed by a 1. Write predicates for each state to prove this claim. Show the theorems that have to be proved. You are not required to prove the theorems.

   Hint: You may use English to describe your predicates, but be unusually precise.

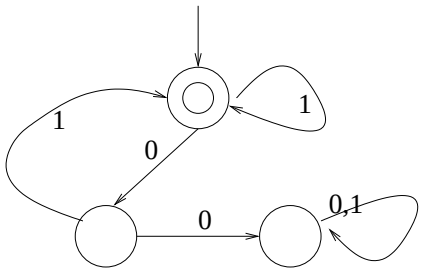

Figure 1: Accept binary strings in which a 0 is immediately followed by a 1

3. (Regular Expressions; 15 points) We would like to specify strings over the alphabet $\{0, 1, 2\}$ which are strictly increasing; thus, 02 is acceptable but 021 and 022 are not.

(a) What is wrong with this solution?

$0^*\ 1^*\ 2^*$

(b) Let *zero* denote the set $\{\epsilon, 0\}$. Write a regular expression for *zero*. Similarly define *one* and *two*. Solve the problem in the first part using these additional symbols in your regular expression.

4. (Types; 15 points) What are the types of the functions defined below? Below `++` concatenates two lists (whose elements are of the same type) to form a single list. So, `[2,3] ++ [4,5,6] = [2,3,4,5,6]`.

   Hint: Do *not* try to understand what each function is computing.

   (a) `charVal n = chr(n + (ord '0'))`. Assume n is of type `Int`.

   (b)
   ```
   parallel ((a,b),(c,d)) ((u,v),(x,y)) =
    (d-b) * (x-u) == (y - v) * (c - a)
   ```
   Assume that a, b, c, d, u, v, x, y are of type `Int`.

   (c) `test f r = (f r) || not (f r)`

   (d) Assume n is of type `Int`.
   ```
   tower n a b c
        | n == 0    = []
        | otherwise =   (tower (n-1) a c b)
                      ++ [(n,a,b)]
                      ++ (tower (n-1) c b a)
   ```

   (e)
   ```
   flatten [] = []
   flatten (xs : xss) = xs ++ (flatten xss)
   ```

5. (Haskell Programming; 15 points)

   (a) Define a function whose type is `[(a,b)] -> ([a],[b])`, where a and b are polymorphic type variables.

   (b) Consider the following variation, $newfib$, of the Fibonacci sequence. The first three items of the sequence are 0, 1 and 2. Any other item in the sequence is a sum of its last three items. Define an efficient procedure for computing $newfib\ n$, for any n, $n \geq 0$.

   (c) Define a function which takes as input a list of integers with at least two elements, and computes the smallest difference $x - y$, where x and y are adjacent elements. Thus, in the list $[3, 7, 2, 4, 8, 11, 9]$, the smallest difference is -4, for $(3, 7)$ and $(4, 8)$.