CMSC330 Spring 2014 Quiz #2

Name _____

Discussion Time	**10am**	**11am**	**noon**	**1pm**	**2pm**
TA Name (circle):	**Tammy**	**Tammy**	**Tammy**	**Daniel**	**Daniel**
		Ilse	**Casey**	**Ian**	

Instructions

- Do not start this test until you are told to do so!
- You have 15 minutes for this quiz.
- This is a closed book exam. No notes or other aids are allowed.
- Answer essay questions concisely in 2-3 sentences. Longer answers are not needed.
- For partial credit, show all of your work and clearly indicate your answers.
- Write neatly. Credit cannot be given for illegible answers.

1. (8 pts) OCaml Types and Type Inference

   a. (2 pts) Give the type of the following OCaml expression

   ```
   fun x y -> (y + 2, x)
   ```
   Type =

   b. (3 pts) Write an OCaml expression with the following type

   ```
   (float list -> float -> 'a) -> 'a
   ```
   Code =

   c. (3 pts) Give the value of the following OCaml expression. If an error exists, describe the error. The function fold is given for problem 2.

   ```
   fold ((fun x y z -> x + (y * z)) 2) 1 [1; 2; 3];;
   ```
   Value =

2. (8 pts) OCaml Programming

Using either map or fold and an anonymous function, write a curried function called **divisible** which when given a number *n* and a list of ints *lst*, returns a list of all elements of *lst* that are divisible by *n* (maintaining their relative ordering). You are allowed to use List.rev (reverses a list) and the (curried) map and fold functions provided, but no other OCaml library functions. **Hint:** x is divisible by y iff (x mod y = 0) is true.

```
let rec map f l = match l with
   [] -> []
   | (h::t) -> (f h)::(map f t)
```

```
let rec fold f a l = match l with
   [] -> a
   | (h::t) -> fold f (f a h) t
```

Example:

divisible 4 [3;16;24] // returns [16; 24]
divisible 3 [4;1;11] // returns [ ]
divisible 3 [ ] // returns [ ]

3. (4 pts) Context Free Grammars

Consider the following grammar:

S → aSc | b | epsilon

a. (2 pts) Describe the set of strings accepted by this grammar.

b. (2 pts) Draw a parse tree for the string aabcc.