

CMSC330 Spring 2012 Quiz #3

Name _____

Discussion Time	10am	11am	12pm	1pm	2pm
TA Name (circle):	Tammy	Tammy	Jane Tim	Jane Richard	Richard

Instructions

- Do not start this test until you are told to do so! You have 20 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, write neatly & show all of your work.

1. (8 pts) OCaml Types and Type Inference

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

`fun x y -> [ x ; y ]` **Type =**

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

`(int -> 'a) -> 'a` **Code =**

- c. (3 pts) Give the value of the following OCaml expression. If an error exists, describe the error. The function map is given on the next page.

`map ((fun x y -> x+y) 1) [2;3;4]` **Value =**

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

2. (14 pts) OCaml Programming

Solve the following OCaml programming problems. The following rules apply to both parts of this question. You are allowed to use `List.rev` (reverses a list) and the (curried) `map` and `fold` functions provided, but no other OCaml library functions. Your solution must run in $O(n)$ time for input lists of length n (note that using `append` instead of `prepend` will usually make your algorithm $O(n^2)$).

- (6 pts) Write a function *growNum* which when given a number k , returns a list of the first k integers, starting from k .

Example:

```
growNum 0 = []
growNum 1 = [1]
growNum 4 = [1;2;3;4]
```

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

- b. (8 pts) Using either map or fold and an anonymous function, write a curried function *countNum* which when given a number *n* and a list of ints *lst*, returns the number of times *n* occurs in *lst*.

Example:

```
countNum 5 [1;2;3;4] = 0
countNum 5 [1;2;3;4;5;6] = 1
countNum 5 [5;5;6;5] = 3
```

3. (8 pts) Context Free Grammars

Consider the following grammar:

$$S \rightarrow aSb \mid aS \mid \text{epsilon}$$

- a. (2 pts) What is the set of strings accepted by this grammar?
- b. (2 pts) Provide a derivation of the string “aab” for this grammar.
- c. (4 pts) Provide a context free grammar for all strings of 0's and 1's representing all odd binary numbers (i.e., binary numbers ending in 1).