

CMSC 330 Fall 2016 Quiz #2 Solution

Name _____

Discussion Time (circle one) 10am 11am 12noon 1pm 2pm 3pm

Discussion TA (circle one) Alex Austin Ayman Brian Damien Daniel K.
Daniel P. Greg Tammy Tim Vitung Will K.

Instructions

- Do not start this quiz until you are told to do so.
- You have 20 minutes for this quiz.
- This is a closed book quiz. No notes or other aids are allowed.
- For partial credit, show all of your work and clearly indicate your answers.

1. (4 points) Without using explicit type declarations, write OCaml expressions of type:

a. `'a list -> 'a -> 'b -> 'a * 'b`

```
let f t x y = match t with
  [] -> (x, y)
  | h::_ -> (h, y);;
```

b. `int -> string -> int * string -> bool`

```
let g x s (x2, s2) =
  (x + 1) = x2 && (s ^ ``.txt') = s2;;
```

2. (4 points) Give the type of `d` in each of the following OCaml expressions:

a. `let d = ((1, 2), [3;0], [])`

`(int * int) * int list * 'a list`

b. `let d = (fun x y z -> (x +. y) > z) 3.14`

`float -> float -> bool`

3. (5 points) Implement a function `insert_at_n ('a list -> 'a -> int -> 'a list)`, which will insert an element at position `n` in a list. If `n` is greater than the length of list, then insert the element to the end. Index starts at 0, and you can assume that `n` is a nonnegative integer. You can write helper functions.

```
let rec insert_at_n lst ele n = match lst with
  [] -> [ele]
  |h::t ->if n = 0 then ele::h::t
          else h::(insert_at_n t ele (n - 1));;
```

4. (7 points) Consider the following OCaml variant type definition for a binary tree:

```
type binary_tree =
  Nil
  | Leaf of int
  | Node of int * binary_tree * binary_tree
```

Write a function called `leaf_sum` that takes as input a `binary_tree` and returns the sum of all of the values of the leaves in the `binary_tree`. You can write helperfunctions.

```
let rec leaf_sum t = match t with
  | Nil -> 0
  | Leaf v -> v
  | Node (_, l, r) -> leaf_sum l + leaf_sum r;;
```