



CMSC330 Spring 2024 Quiz 2

Proctoring TA: _____ Name: _____

Section Number: _____ UID: _____

Problem 1: Basics

[Total 4 pts]

	True	False
Regular expressions can be used to parse text out of strings	<input type="radio"/>	<input type="radio"/>
When evaluating an expression, the order matters when there are side effects	<input type="radio"/>	<input type="radio"/>
The concept of fold is limited to lists	<input type="radio"/>	<input type="radio"/>
Map cannot be written in terms of fold_left	<input type="radio"/>	<input type="radio"/>

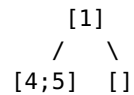
Problem 2: Data and Map

[Total 8 pts]

Consider the following Variant from project 2:

```
type 'a tree = Leaf|BiNode of 'a tree * 'a * 'a tree (* left subtree, value, right subtree *)
```

Suppose we want to make a tree that looks like:



(a) How would you create a variable called `t` that is bound to a `int list tree` that corresponds to the above tree? [3 pts]

(b) Tree Map

[5 pts]

Suppose we have a function called `tree_map`. It works like `map`, but will map a `'a tree` to `'b tree`. Using only `tree_map` and `map`, write a function that will add 5 to every element of the lists within a `int list tree`.

```
val tree_map f t: ('a -> 'b) -> 'a tree -> 'b tree
val map f l: ('a -> 'b) -> 'a list -> 'b list
let addfive ltree =
```

Problem 3: Regex

[Total 8 pts]

*	zero or more repetitions of the preceding character or group
+	one or more repetitions of the preceding character or group
?	zero or one repetitions of the preceding character or group
.	any character
$r_1 r_2$	r_1 or r_2 (eg. $a b$ means 'a' or 'b')
[abc]	match any character in abc
[$\sim r_1$]	anything except r_1 (eg. [$\sim abc$] is anything but an 'a', 'b', or 'c')
[r_1-r_2]	range specification (eg. [a-z] means any letter in the ASCII range of a-z)
{n}	exactly n repetitions of the preceding character or group
{n,}	at least n repetitions of the preceding character or group
{m,n}	at least m and at most n repetitions of the preceding character or group
^	start of string
\$	end of string
(r_1)	capture the pattern r_1 and store it somewhere (match group in Python)
\d	any digit, same as [0-9]
\s	any space character like \n, \t, \r, \f, or space

Write a regex that describes a subset of valid umd emails. Emails take the form of a user's directory ID followed by the @ symbol, followed by one of the following domain names: cs.umd.edu, terpmail.umd.edu, or just umd.edu.

- A user's directory ID can be length 0 to length 8 consisting of only alphanumeric (both upper and lowercase) characters.
- A user's directory ID may not start with a digit.

(a) Email Addresses

[4 pts]

(b) Assuming a full match, which strings are accepted by the following regex? Select all that apply

[2 pts]

$^{[Cliff|Anwar]^+ is (great|the best)?}$

- (A) Cliff is sad
 (B) Anwar is the best
 (C) cliff is great
 (D) winwrar is
 (E) flan is the best
 (F) the best

(c) Which of the following regular expressions is not equivalent to the others?

[2 pts]

- (A) $[abc]^+def?(g|hi)$
 (B) $(a|b|c)[abc]^*def?g|(a|b|c)[abc]^*def?hi$
 (C) $(abc)^+def(g|hi)|(abc)(abc)^*de(g|hi)$
 (D) $[abc]^+de((g|hi)|fg|fhi)$
 (E) They are all the same