

CMSC330 Fall 2024 Quiz 2



Proctoring TA: _____ Name: _____

Section Number: _____ UID: _____

Problem 1: Basics

[Total 4 pts]

Regular expressions can describe strings with 2 sets of balanced parenthesis

True

False

In the expression `let x = ref 4 in let y = x in !y`, both `x` and `y` point to the same thing

The **concept** of fold is limited to lists

The functions `let f () = print_int 3` and `let f = print_int 3` have the same behavior when called

Problem 2: Data types and Map

[Total 6 pts]

Consider the following Variant:

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

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 `fold_left`, write a function `even_sums` that takes in an `int list tree` and returns a `bool tree`. Each node in the output tree should represent if the sum of the input node is even.

You can write additional helper functions, **but may not use the `rec` keyword**

```
val tree_map f t: ('a -> 'b) -> 'a tree -> 'b tree
val fold_left f a l: ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a
```

```
even_sums (BiNode([], BiNode([1;3;5], Leaf, Leaf), BiNode([2;1;7], Leaf, Leaf)))
=> (BiNode(true, BiNode(false, Leaf, Leaf), BiNode(true, Leaf, Leaf)))
```

```
      []                true
     /  \              /  \
[1;3;5] [2;1;7]    false true
```

```
let even_sums t =
```

Problem 3: Regex

[Total 6 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
[\wedge r ₁]	anything except r_1 (eg. [\wedge 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
\wedge	start of string
\$	end of string

Write a regex that describes **exactly** the room names found in CS related buildings. A room name will have

- the building code (**only** IRB or AVW)
- followed by the room number (any 4 digit number from 0000 to 4500 (inclusive))
- followed by a space and the purpose which is either the last name of the professor or "TA SPACE")
- Last names of all professors will **begin** with any capital letter and have **at least** 3 lowercase letters following their first letter

valid room names

IRB2238 Baka
IRB4500 Mamat
AVW4165 TA SPACE

invalid room names

HJP1206This
IRB2248 bakalian
avw123 Bad

Problem 4: Property Based Testing

[Total 4 pts]

Consider the following function which has a bug in it:

```
1 (* signed_square should take in an int x, square it, but keep the original sign *)
2 let signed_square x = if x < 0 then (-x) * (-x) else x * x
```

Consider the following property: *the output of signed_square should be greater than or equal to the input*

Is this a valid property? Yes/No: Y N

Is the function `fun x -> signed_square x >= x` a correct representation of the property? Yes/No: Y N

If we test this property on the provided code, will it ever return false?

Yes: Y The property is not valid so the result of testing this property is meaningless: NA

No: N