

CECS 424
Assignment 11
Total: 25 Points

General Instruction

- Allowed submission file type for the questions: PDF only
 - I recommend that you type your answers to exercise questions by using a word processor (Microsoft word, LibreOffice writer, L^AT_EX, etc.).
 - This is **not** a group assignment.
 - Submit your work via BeachBoard (Not email or in class).
-

1. (5 points) The `myfoldr` and `mylengthr` are defined in Haskell as follows:

```
myfoldr :: (a -> b -> b) -> b -> [a] -> b
myfoldr f acc []      = acc
myfoldr f acc (x:xs) = f x (myfoldr f acc xs)
```

```
mylengthr :: [a] -> Int
mylengthr = myfoldr (\_ n -> 1 + n) 0
```

Show the evaluation steps of `mylengthr [1,2,3] => ... => 3`.

2. The `myfoldl` is defined in Haskell as follows:

```
myfoldl :: (a -> b -> a) -> a -> [b] -> a
myfoldl f acc []      = acc
myfoldl f acc (x:xs) = myfoldl f (f acc x) xs
```

- (a) (5 points) Write a function called `mylengthl` using `myfoldl`. The `mylengthl` should output the length of a given list.
- (b) (5 points) Show the evaluation steps of `mylengthl [1,2,3] => ... => 3`.
3. The reverse of a list can be computed by using the folding left function.
- (a) (5 points) Write a function called `myreverse` using `myfoldl`. The `myreverse` should output the reverse of a given list.
- (b) (5 points) Show the evaluation steps of `myreverse [1,2,3] => ... => [3,2,1]`.