Bryan Duong

013104973

CECS 424 Assignment 11

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`.

```
myfoldr (\_ n → 1 + n) 0 [1,2,3]
= (\_ n → 1 + n) 1 (myfoldr (\_ n → 1 + n) 0 [2,3])
= (\_ n → 1 + n) 1 ((\_ n → 1 + n) 2 (myfoldr (\_ n → 1 + n) 0 [3]))
= (\_ n → 1 + n) 1 ((\_ n → 1 + n) 2 (((\_ n → 1 + n) 3 0))
= (\_ n → 1 + n) 1 ((\_ n → 1 + n) 2 1)
= (\_ n → 1 + n) 1 2
= 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`.

(a) mylengthl = myfoldl (\n _ → n + 1) 0

(b) mylengthl [1,2,3]

```
= myfoldl (\n _ → n+1) 0 [1,2,3]
= myfoldl (\n _ → n+1) ((\n _ → n+1) 0 1) [2, 3]
= myfoldl (\n _ → n+1) ((\n _ → n+1) ((\n _ → n+1) 0 1 ) 2)) [3]
= ((\n _ → n+1) ((\n _ → n+1) ((\n _ → n+1) 0 1 ) 2) 3)
= ((\n _ → n+1) ((\n _ → n+1) 1 2) 3)
= ((\n _ → n+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]`.

(a) myreverse = myfoldl (\n m → m : n) [ ]

(b) myreverse [1,2,3]
 = myfoldl (\n m → m : n) [ ] [1,2,3]
 = myfoldl (\n m → m : n) ((\n m → m : n) [ ] 1) [2,3]
 = myfoldl (\n m → m : n) ((\n m → m : n) ((\n m → m : n) [ ] 1) 2) [3]
 = ((\n m → m : n) ((\n m → m : n) ((\n m → m : n) [ ] 1) 2) 3)
 = ((\n m → m : n) ((\n m → m : n) [1] 2) 3)
 = ((\n m → m : n) [2,1] 3)
 = [3,2,1]