Man Tik Li
CS 360 – 002
Krzysztof Nowak
August 7, 2020

Lab 3

OS: macOS X Catalina ver 10.15.6
Serial Number (system): C02NLN90G3QJ
Hardware UUID: 793DBFBA-8B38-5104-9E39-7BBF399D3196
Tested ghci version 8.6.5 and implemented in Tux.
*Problem 4 unable to load on Tux, it tested in my machine (ghci, version 7.10.2)
Path Files for all Tux files "/home/ml3546/CS360/lab_3".

1.

i.

```
ml3546@tux5:~/CS360/lab_3_code$ ls
primes.hs  pyth.hs
ml3546@tux5:~/CS360/lab_3_code$ ghci
GHCi, version 8.6.5: http://www.haskell.org/ghc/  :? for help
Prelude> :load pyth.hs
[1 of 1] Compiling Main             ( pyth.hs, interpreted )
Ok, one module loaded.
*Main> pyth 17
[(3,4,5),(5,12,13),(6,8,10),(8,15,17),(9,12,15)]
```

ii.

```
ml3546@tux2:~/CS360/lab_3$ ls
primes.hs  pyth.hs
ml3546@tux2:~/CS360/lab_3$ ghci
GHCi, version 8.6.5: http://www.haskell.org/ghc/  :? for help
Prelude> :load primes.hs
[1 of 1] Compiling Main             ( primes.hs, interpreted )
Ok, one module loaded.
*Main> take 50 tprimes
[3,5,7,11,13,17,19,29,31,41,43,59,61,71,73,101,103,107,109,137,139,149,151,179,181,19
1,193,197,199,227,229,239,241,269,271,281,283,311,313,347,349,419,421,431,433,461,463
,521,523,569]
```

iii. It would not be able to verify that twin primes contain finite items because the number is continually growing. It is challenging to keep track the twin primes have reached the end. If using the sieve of Eratosthenes when marking all the divisible numbers, it will keep getting more significant.

2.

```
ml3546@tux2:~/CS360/lab_3$ cat p2.hs
p2 n = take n [2^k|k <- [0,1..]]
ml3546@tux2:~/CS360/lab_3$ ghci
GHCi, version 8.6.5: http://www.haskell.org/ghc/  :? for help
Prelude> :load p2.hs
[1 of 1] Compiling Main             ( p2.hs, interpreted )
Ok, one module loaded.
*Main> p2 3
[1,2,4]
*Main> p2 4
[1,2,4,8]
*Main> p2 5
[1,2,4,8,16]
*Main> p2 6
[1,2,4,8,16,32]
*Main> p2 7
[1,2,4,8,16,32,64]
*Main> p2 8
[1,2,4,8,16,32,64,128]
*Main> p2 9
[1,2,4,8,16,32,64,128,256]
*Main> p2 10
[1,2,4,8,16,32,64,128,256,512]
*Main> p2 11
[1,2,4,8,16,32,64,128,256,512,1024]
*Main> p2 12
[1,2,4,8,16,32,64,128,256,512,1024,2048]
*Main> p2 13
[1,2,4,8,16,32,64,128,256,512,1024,2048,4096]
*Main> p2 14
[1,2,4,8,16,32,64,128,256,512,1024,2048,4096,8192]
*Main>
```

3. Please see insert.hs

```
(base) Tiks-MacBook-Pro-133-Retina:Downloads TIK_MONICA$ cat church.hs
overrideZeroC = const id
overrideMultiC = (.)
overrideAddC = (<*>) . fmap (.)

overrideSucc = (<*>) (.)
getChurchIntValue :: Int -> ((a -> a) -> a -> a)
getChurchIntValue 0 = overrideZeroC
getChurchIntValue n = overrideSucc $ getChurchIntValue (n - 1)


getIntValue :: ((Int -> Int) -> Int -> Int) -> Int
getIntValue x = x succ 0

[cFive] = getChurchIntValue <$> [5]

churchAdd =  print $  getIntValue <$> [overrideAddC cFive cFive]
churchMulti = print $  getIntValue <$> [overrideMultiC cFive cFive]
```

4.

```
(base) Tiks-MacBook-Pro-133-Retina:Downloads TIK_MONICA$ ghci
GHCi, version 7.10.2: http://www.haskell.org/ghc/  :? for help
Prelude> :load church.hs
[1 of 1] Compiling Main             ( church.hs, interpreted )
Ok, modules loaded: Main.
*Main> churchAdd
[10]
*Main> churchMulti
[25]
```