AllenbyExercises1.2

February 25, 2025

[ ]:
```
Introduction to Number Theory with Computing, Allenby and Redfern,
```

Pages 19-24 Sieve of Eratosthenes

[1]:
```
integers = [2..] -- an infinite list, starting with 2 and increasing in steps of␣
↪1
take 5 integers -- returns a list of its first 5 elements
```

```
[2,3,4,5,6]
```

[2]:
```
4 `mod` 2 -- 4 modulo 2
```

```
0
```

$ s = { 2*x ; |;; x ∈; integers, x ≠ 0; mod; 2}$asetgenerator

[3]:
```
s = [2*x | x <- integers, odd x] -- an equivalent Haskell list comprehension
```

[4]:
```
headElements = take 5 s
headElements
```

```
[6,10,14,18,22]
```

[5]:
```
-- function that returns the head of a list x:xs
-- (where : is the list constructor "cons" and xs is the rest of the list)
head (x:xs) = x
head headElements
```

```
6
```

[55]:
```
-- Implementation of BASIC program 1.2 page 23
primes :: [Int] -- primes is a list of Ints
primes = sieve [2..]
  where
-- peel off head of argument and apply sieve function to the rest
    sieve (p:xs) = p : sieve [x | x <- xs, x `mod` p /= 0]
```

[7]:
```
take 50 primes
```

```
[2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,
97,101,103,107,109,113,127,131,137,139,149,151,157,163,167,173,
179,181,191,193,197,199,211,223,227,229]
```

Exercise 1.2 (1)

[56]:
```
-- We will calculate the distance between successive primes and test when that is 6 or more
p = primes
t = tail primes
diff (u, v) = v - u
```

[16]:
```
-- Lemma calculate fact 7
fact :: Int -> Int
fact 1 = 1
fact n = n * fact (n-1)
```

[17]:
```
fact 7
```

```
5040
```

[9]:
```
test = zip p t
```

[24]:
```
-- locate the primes near to 5040
map (<5040) (take 700 primes)
```

```
[True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
```

```
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
rue,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,True,
True,True,True,False,False,False,False,False,False,False,False,False,False,False,
False,False,False,False,False,False,False,False,False,False,False,False,False,False]
```

[34]:
```
test !! 674
```

```
(5039,5051)
```

[ ]:
```
-- to get to fact 7 we need 674 primes
map diff (take 674 test)
```

```
[1,2,2,4,2,4,2,4,6,2,6,4,2,4,6,6,2,6,4,2,6,4,6,8,4,2,4,2,4,14,4,6,2,10,2,6,6,4,6,6,2,
10,2,4,2,12,12,4,2,4,6,2,10,6,6,6,2,6,4,2,10,14,4,2,4,14,6,10,2,4,6,8,6,6,4,6,8,4,
8,10,2,10,2,6,4,6,8,4,2,4,12,8,4,8,4,6,12,2,18,6,10,6,6,2,6,10,6,6,2,6,6,4,2,12,10,
2,4,6,6,2,12,4,6,8,10,8,10,8,6,6,4,8,6,4,8,4,14,10,12,2,10,2,4,2,10,14,4,2,4,14,4,2,
4,20,4,8,10,8,4,6,6,14,4,6,6,8,6,12,4,6,2,10,2,6,10,2,10,2,6,18,4,2,4,6,6,8,6,6,22,2,
10,8,10,6,6,8,12,4,6,6,2,6,12,10,18,2,4,6,2,6,4,2,4,12,2,6,34,6,6,8,18,10,14,4,2,4,6,
8,4,2,6,12,10,2,4,2,4,6,12,12,8,12,6,4,6,8,4,8,4,14,4,6,2,4,6,2,6,10,20,6,4,2,24,4,2,
10,12,2,10,8,6,6,6,18,6,4,2,12,10,12,8,16,14,6,4,2,4,2,10,12,6,6,18,2,16,2,22,6,8,
6,4,2,4,8,6,10,2,10,14,10,6,12,2,4,2,10,12,2,16,2,6,4,2,10,8,18,24,4,6,8,16,2,4,8,
16,2,4,8,6,6,4,12,2,22,6,2,6,4,6,14,6,4,2,6,4,6,12,6,6,14,4,6,12,8,6,4,26,18,10,8,4,
6,2,6,22,12,2,16,8,4,12,14,10,2,4,8,6,6,4,2,4,6,8,4,2,6,10,2,10,8,4,14,10,12,2,6,4,
2,16,14,4,6,8,6,4,18,8,10,6,6,8,10,12,14,4,6,6,2,28,2,10,8,4,14,4,8,12,6,12,4,6,20,
10,2,16,26,4,2,12,6,4,12,6,8,4,8,22,2,4,2,12,28,2,6,6,6,4,6,2,12,4,12,2,10,2,16,2,
16,6,20,16,8,4,2,4,2,22,8,12,6,10,2,4,6,2,6,10,2,12,10,2,10,14,6,4,6,8,6,6,16,12,
2,4,14,6,4,8,10,8,6,6,22,6,2,10,14,4,6,18,2,10,14,4,2,10,14,4,8,18,4,6,2,4,6,2,12
,4,20,22,12,2,4,6,6,2,6,22,2,6,16,6,12,2,6,12,16,2,4,6,14,4,2,18,24,10,6,2,10,2,
10,2,10,6,2,10,2,10,6,8,30,10,2,10,8,6,10,18,6,12,12,2,18,6,4,6,6,18,2,10,14,6,
4,2,4,24,2,12,6,16,8,6,6,18,16,2,4,6,2,6,6,10,6,12,12,18,2,6,4,18,8,24,4,2,4,6,2,
12,4,14,30,10,6,12,14,6,10,12,2,4,6,8,6,10,2,4,14,6,6,4,6,2,10,2,16]
```

[ ]:
```
-- where is the first distance greater than 6
import Data.List
elemIndex 8 it
```

```
Just 23
```

[ ]:
```
-- the least k is 91
test !! 23
```

```
(89,97)
```

Exercise 1.2 (3) Allenby

[37]:
```
isPrime u = u `elem` take 100 p
```

[38]:
```
isPrime 91
```

```
False
```

[40]:
```
facts = map fact [1..]
```

[42]:
```
take 7 facts
```

```
[1,2,6,24,120,720,5040]
```

[44]:
```
pairInts n = (n-1,n+1)
```

[45]:
```
map pairInts it
```

```
[(0,2),(1,3),(5,7),(23,25),(119,121),(719,721),(5039,5041)]
```

[46]:
```
testPrimality (u,v) = (isPrime u, isPrime v)
```

[47]:
```
map testPrimality it
```

```
[(False,True),(False,True),(True,True),(True,False),(False,False),(False,False),(False,False)]
```

Exercise 1.2 (4) Allenby

[51]:
```
pairIntsStar n = (fact n - n - 1,fact n + n + 1)
```

[52]:
```
map pairIntsStar [1..7]
```

```
[(-1,3),(-1,5),(2,10),(19,29),(114,126),(713,727),(5032,5048)]
```

[53]:
```
map testPrimality it
```

```
[(False,True),(False,True),(True,False),(True,True),(False,False),(False,False),(False,False)]
```

Exercise 1.2 (4) Allenby Three successive odd numbers are 3k,3k+1,3k+2 or 3k-1,3k,3k+1 or 3k-2,3k-1,3k for some k. In any sequence there is a multiple of 3.

[ ]:
```

```