Lauro Cabral

February 7, 2018

CECS 229

HW2

Section 4.3

2) Determine whether each of these integers is prime.

a) 19

```
19 is a prime number
```

b) 27

```
27 is not a prime number because 27 / 3 = 9
```

c) 93

```
93 is not a prime number because 93 / 3 = 31
```

d) 101

```
101 is a prime number
```

e) 107

```
107 is a prime number
```

f) 113

```
113 is a prime number
```

4) Find the prime factorization of each of these integers

a) 39

```
39 = 13 x 3
```

b) 81

```
81 / 3 = 27
27 / 3 = 9
9  / 3 = 3
3  / 3 = 1
81 = 3x3x3x3⇔3^4
```

c) 101

```
101 / 1 = 101
```

d) 143

```
143 = 11(13)
```

e) 289

```
289 = 17(17) = 17^2
```

f) 899

899 = 29 / 31

10) Show that if 2^m+1 is an odd prime, then $m=2^n$ for some nonnegative integer *n*.

t is odd and m = kt

$x^{kt} + 1 = (x^k+1)(x^{k(t-1)}-x^{k(t-2)}+ \dots - x^k+1)$

$(x^k+1)(x^{k(t-1)}-x^{k(t-2)}+ \dots - x^k+1) = x^k(x^{k(t-1)}-x^{k(t-2)}+ \dots - x^k+1)+1(x^{k(t-1)}-x^{k(t-2)}+ \dots - x^k+1)$

$= (x^{k(t-1)+k}-x^{k(t-2)+k}+ \dots - x^{k+k}+1(x^k)) + (x^{k(t-1)}-x^{k(t-2)}+ \dots - x^k+1)$

$= (x^{k(t)}-x^{k(t-1)}+ x^{k(t-2)} \dots - x^{2k} + x^k)$

$\underline{+ (x^{k(t-1)}-x^{k(t-2)}+ \dots + x^{2k} - x^k + 1)}$

$X^{kt}+1$

18) We call a positive integer **perfect** if it equals the sum of its positive divisors other than itself.

a) Show that 6 and 28 are perfect.

Divisors for 6 are 1,2,3,6
Divisors for 28 are 1,2,4,7,14,28
6: 1 + 2 + 3 = 6 , so 6 is perfect
28: 1 + 2 + 4 + 7 + 14 = 28, so 28 is perfect

b) Show that $2^{p-1}(2^p-1)$ is a perfect number when 2^p-1 is prime.

The divisor for $2^{p-1}(2^p-1)$ other than itself is: all powers of 2 up to 2^{p-1} which is $2^0,2^1,2^2 \dots 2^{p-1}$

20) Determine whether each of these integers is prime, verifying some of Mersenne's claim.

a) 2^7-1

$2^7-1 = 127$
$\sqrt{127} = 11.26$, prime numbers less than 11.23 are 2,3,5,7,11
Since 127, is not divisible by those numbers means 2^7-1 is a prime number

b) 2^9-1

$2^9-1 = 511$
$\sqrt{511} = 22.60$, prime numbers less than 22.60 are 2,3,5,7,11,13,14,19
Since 511, is divisible by 7 numbers means 2^9-1 is not a prime number

c) $2^{11}-1$

1. $2^{11}-1 = 2047$
2. $\sqrt{2047} = 45.24$, prime numbers less than 45.24 are 2,3,5,7,11,13,14,19,23,29,31,37,41,43
3. Since 2047, is divisible by 23 numbers means $2^{11}-1$ is not a prime number

d) $2^{13}-1$

1. $2^{13}-1 = 8191$
4. $\sqrt{8191} = 90.50$, prime numbers less than 90.50 are 2,3,5,7,11,13,14,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79
2. Since 8191, is not divisible means $2^{13}-1$ is a prime number

24) What are the greatest common divisors of these pairs of integers?

a) $2^2*3^3*5^5$, $2^5*3^3*5^2$

1. $gcd(2^2*3^3*5^5,2^5*3^3*5^2)= 2^2*3^3*5^2$

b) $2*3*5*7*11*13$, $2^{11}*3^9*11*17^{14}$

1. $gcd(2*3*5*7*11*13,2^{11}*3^9*11*17^{14})= 2*3*5*11$

c) 17, 17^{17}

1. $gcd(17,17^{14})=17$

c) 2^2*7, 5^3*13

1. $gcd(2^2*7, 5^3*13)=1$

d) 0, 5

1. gcd(0, 5)=5

e) 2*3*5*7, 2*3*5*7

1. gcd(2*3*5*7, 2*3*5*7)=2*3*5*7

28) Find gcd(1000,625) and lcm(1000,625) and verify that gcd(1000,625) * lcm(1000,625) = 1000*625.

```
1.  1000 = 2(500)        625 = 5(125)
2.   500 = 2(250)        125 = 5(25)
3.   250 = 2(125)         25 = 5(5)
4.   125 = 5(25)           5 = 5(1)
5.    25 = 5(5)          625 = 5^4
6.     5 = 5(1)
7.  1000 = 2^3*5^3
8.
9.  gcd(2^3*5^3,5^4) = 5^3     lcm(2^3*5^3,5^4) = 2^3*5^4
10. 5^3*2^3*5^4 = 625000
11.
```

30) If the product of two integers is $2^7 3^8 5^2 7^{11}$ and their greatest common divisor is $2^3 3^4 5$, what is their least common multiple?

Product = gcd * lcm

lcm = Product / gcd

$\text{lcm} = \frac{2^7 3^8 5^2 7^{11}}{2^3 3^4 5} = 2^4 3^4 5^1 7^{11}$

GCD

1) Prime Factorization

```
# Prime Factorization Algorithm
def primeFactors(n):
  d = 2
  factors = []# empty list
  while n > 1:
    if n % d == 0:
      factors.append(d) n = n / d
    else:
      d = d + 1
return factors
#Euclidean Algorithm
def pgcd(a, b):
  i = 0
  x = primeFactors(a)
  y = primeFactors(b)
  print(x)
  print(y)
  while i < len(x):
   z = 0
  return z

s = pgcd(315, 13)
print(s)
```

2) Euclidean

```
#Euclidean Algorithm
def gcd(a, b):
  x = a y = b
  while (y != 0):
    r = x % y
    x = y
    y = r
  return x

s = gcd(100, 111)
print(s)
```