List of Mathematica Commands Useful for Number Theory

NOTE: On a MacIntosh commands are entered by hitting the enter key. On machines with Unix a command is entered by hitting the return key.

Basic Mathematica Commands

**BaseForm[n,b]**: Gives the base b expansion of n.

**ChineseRemainder[{a,b},{m,n}]**: Gives the unique solution to the congruences \( x \equiv a \pmod{m} \), \( x \equiv b \pmod{m} \) when \( \gcd(m,n) = 1 \).

**Divisors[n]**: Gives a list of integers that divide n.

**EulerPhi[n]**: Gives the value of the Euler Phi Function \( \phi(n) \).

**ExtendedGCD[m,n]**: Gives \( \{d,\{r,s\}\} \) where \( d \) is the greatest common divisor of \( m \) and \( n \) and \( r,s \) satisfy \( d = mr + ns \).

**GCD[m,n]**: Gives the greatest common divisor of \( m \) and \( n \). Can also be used with more than two integers.

**FactorInteger[n]**: Gives the factorization of n into a product of prime numbers. \( \{p,b\} \) indicates a factor of \( p^b \).

**FactorIntegerECM[n]**: Gives a factor (possibly composite) of an integer rather than completely factoring it.

**IntegerExponent[n,p]**: Gives the exponent of prime \( p \) in the factorization of n.

**JacobiSymbol[n,m]**: Gives the Jacobi symbol \( \left( \frac{n}{m} \right) \) which is the same as the Legendre symbol when m is an odd prime.

**LCM[m,n]**: Gives the least common multiple of \( m \) and \( n \). Can also be used with more than two integers.

**Mod[a,m]**: Gives \( a \pmod{m} \).

**MultiplicativeOrder[a,n]**: Gives the order of \( a \pmod{n} \).

**NextPrime[n]**: Gives the smallest prime number greater than n.

**PowerMod[a,b,m]**: Gives \( a^b \pmod{m} \) for large \( b \) when the command \( \text{Mod}[a^b,m] \) will not. The special case \( b = -1 \) will give the inverse of \( a \pmod{m} \) if it exists.

**PrimitiveRoot[n]**: Gives a primitive root modulo n when \( n = p \) or \( n = 2p \), where \( p \) is a prime.

**Prime[n]**: Gives the \( n \)th prime number.

**PrimeQ[n]**: Gives True if \( n \) is prime, False if \( n \) is not prime.

**Quotient[a,b]**: Gives the integer part of a divided by b.

**Quit**: terminates a Mathematica session.

**Random[Integer,\{min,max\}]**: Gives a random integer between \( \text{min} \) and \( \text{max} \).

**SqrtMod[d,n]**: Gives the square root of \( d \) modulo \( n \) when it exists.

**SquareFreeQ[n]**: Returns True if \( n \) contains a squared factor and False if not.

More Advanced Mathematica Commands

More advanced commands are available by loading what are called Packages into any Mathematica session you are running. There are several Number Theory packages. One of interest to us is the PrimalityProving package. This package gives commands which tell you whether or not Mathematica is sure a number is prime or composite, and gives a proof (which it calls a certificate) that the number is prime or composite when it is sure.
This can be loaded by typing:
<< PrimalityProving'' (Note: there is an apostrophe at the end).

After you have tried to load a package you can make sure it is loaded by typing:

\$Packages: Lists packages currently loaded.

After loading the PrimalityProving package one has commands that allows one to

**ProvablePrimeQ[n]**: Gives True if \( n \) can be proved to be prime and False if \( n \) can be
proved to be composite. It will not return an answer if it doesn’t know for sure. This
command is slower than the PrimeQ command, so if a number is small it is better to use
the PrimeQ command.

**PrimeQCertificate[n]**: Prints a certificate that proves \( n \) is prime or composite. A cer-
tificate is a set of data that indicates why \( n \) is prime or composite.

A proof that the number is composite will consist of three numbers, in the form of a)
or b) below

a) \( \{a, n - 1, n\} \) which means

\[ a^{n-1} \not\equiv 1 \pmod{n} \]

b) \( \{a, 2, n\} \) which means

\[ a^2 \equiv 1 \pmod{n} \]

We will learn why these facts determine the number is composite.

The proof that a number is prime is much more complicated. It uses the theory of
elliptic curves, which is beyond the scope of our course.