

Computer Tutorial 10

1. The MAGMA function `PrimesUpTo` returns the set of prime numbers up to a specified number. So `PrimesUpTo(1000)` is the set of primes less than 1000. Print this out, and then use commands such as `#PrimesUpTo(1000)` to find the number of primes less than 1000, 2000, \dots , 10000.
2. Let $\pi(n)$ be the number of primes less than the positive integer n . The famous Prime Number Theorem (hard to prove!) states that $\pi(n) \sim \frac{n}{\ln n}$ as $n \rightarrow \infty$. That is, $\pi(n) \ln(n)/n \rightarrow 1$ as $n \rightarrow \infty$ (but the convergence is slow). Compute $\pi(n) \ln(n)/n$ for $n = 10^i$, for each i from 1 to 8. (The MAGMA function `Log`, when applied to a real number x , returns $\ln(x)$. So you can use `#PrimesUpTo(1000)*Log(1000)/1000`; etc.)
3. The best primes p for the Elgamal cryptosystem are those for which $\frac{1}{2}(p-1)$ is also prime. (These are called *safe primes*). Find all the safe primes less than 1000, and the number of safeprimes less than 1000, 2000, \dots , 10000. (One solution is as follows:

```
safeprimes:= function(N)
  SP:={};
  p:=3;
  while p le N do
    if IsPrime((p-1) div 2) then
      Include(~SP,p);
    end if;
    p:=NextPrime(p);
  end while;
  return SP;
end function;
```

after which you can use `safeprimes(1000)`; `#safeprimes(5000)`; etc.)

(I found that there are 664579 primes less than 10000000, and 30657 of them are safe.)

4. You can interpret the Prime Number Theorem as saying then the probability that a random number less than n is prime is about $1/\ln(n)$, for n large enough. If the events that n is prime and that $(n-1)/2$ is prime were independent, the probability of them both occurring simultaneously would be the product of their separate probabilities, about $(1/\ln n)(1/\ln(n/2))$. So the number of safe primes less than n would be about $n(1/\ln n)(1/\ln(n/2))$. Compute this for $n = 1000, 2000, \dots, 10000$, and compare with the numbers you found in Exercise 3.

(The estimate $n(1/\ln n)(1/\ln(n/2))$ suggested above is too large; a more sophisticated argument suggests that a better guess is $Cn/(\ln(n/2))^2$, where C is a constant—called the “twin prime constant”—defined to be $\prod_{p \geq 3} \frac{p(p-2)}{(p-1)^2}$, the product being over all odd prime numbers. In fact $C \approx 0.66016$. However, it has not even been proved that there are an infinite number of primes p for which $\frac{1}{2}(p-1)$ is also prime).

5. Get MAGMA to give you an approximation to C :
`C:=RealField()!(&*[p*(p-2)/(p-1)^2 : p in PrimesUpTo(10000) | p gt 2])`;
`C`;

6. Put $n = 10^{300}$, and assume that $Cn/(\ln(n/2))^2$ approximates the number of safe primes less than n . So the probability that a randomly chosen number less than n is a safe prime is $C/(\ln(n/2))^2$. Now imagine choosing k random numbers less than n , in the hope that one of them is a safe prime. How large does k have to be to make the probability of success greater than 0.5? (Hint: the probability that none are safe primes is $(1-t)^k$, where t is the probability that a random number less than n is a safe prime.)
7. In MAGMA's terminology, if p is a prime then `FiniteField(p)` is the set $\{0, 1, \dots, p-1\}$, with addition and multiplication defined to be the same as ordinary addition and multiplication followed by reduction modulo p . MAGMA's "coercion operator" `!` can be used to convert an integer to an element of `FiniteField(p)`, or vice versa. Type `F:=FiniteField(97);` and now put `a:=F!50; b:=F!77;`. Check that `a*b`; and `a+b`; agree with `50*77 mod 97`; and `(50+77) mod 97`; . Similarly, check that `a^111` agrees with `Modexp(50,111,97);`, and that `b^(-1)` agrees with `InverseMod(77,97);`.

8. We now want to do some calculations with polynomials, working modulo 97. Type

```
P<x>:=PolynomialRing(F);
f5:=(x-1)*(x-2)*(x-3)*(x-4)/((5-1)*(5-2)*(5-3)*(5-4));
f4:=(x-1)*(x-2)*(x-3)*(x-5)/((4-1)*(4-2)*(4-3)*(4-5));
f3:=(x-1)*(x-2)*(x-4)*(x-5)/((3-1)*(3-2)*(3-4)*(3-5));
f2:=(x-1)*(x-3)*(x-4)*(x-5)/((2-1)*(2-3)*(2-4)*(2-5));
f1:=(x-2)*(x-3)*(x-4)*(x-5)/((1-2)*(1-3)*(1-4)*(1-5));
```

Observe that the polynomial `f5` should take the value 0 when x is given the value 1, or 2, or 3, or 4, and 1 when x is given the value 5. Check using `Evaluate(f5,1);` etc.. Or check them all at once with

```
[ [ Evaluate(fi,xj) : xj in [1..5] ] : fi in [f1,f2,f3,f4,f5] ];
```

Now define `a1:=73; a2:=50; a3:=36; a4:=82; a5:=17;` and

```
f:=a1*f1+a2*f2+a3*f3+a4*f4+a5*f5;
```

What values will `f` take at 1, 2, 3, 4 and 5? Check it.

9. Continuing with `f1, \dots, f5` as defined in Exercise 8, define

```
g:=33+Random(F)*x+Random(F)*x^2+Random(F)*x^3+Random(F)*x^4;
a1:=Evaluate(g,1);
a2:=Evaluate(g,2);
a3:=Evaluate(g,3);
a4:=Evaluate(g,4);
a5:=Evaluate(g,5);
```

and check that `a1*f1+a2*f2+a3*f3+a4*f4+a5*f5` equals `g`.

10. `S` is a Very Important And Secret Number. Five honest people, p_1, p_2, p_3, p_4 and p_5 have each been given partial information. Specifically, they have been told that `S` is less than the prime `p`, which is 447555834974539, and that `S` is the constant term of a quadratic polynomial $f(x)$ over `FiniteField(p)`. In addition, person p_i is told the value of $f(i)$. Unfortunately, p_2 and p_5 are killed. So p_1, p_3 and p_4 hold an urgent meeting, and share their information: p_1 's number is 196231291191342, p_3 's is 195412581909834, and p_4 's is 163633523397347. Find `S`.

(Put `F:=FiniteField(447555834974539)` and `P<x>:=PolynomialRing(F)`, and define polynomials `f4, f3` and `f1` by `f4:=(x-1)*(x-3)/((4-1)*(4-3));` etc.. Then put

```
f:=196231291191342*f1+195412581909834*f3+163633523397347*f4;
```

and use `Evaluate(f,0);`)

(Note that one could alternatively calculate `S` by means of the following formula:

```
S:=196231291191342*m1+195412581909834*m3+163633523397347*m4;
```

where `m4:=(0-1)*(0-3)/((4-1)*(4-3));` etc.)