# Procedures for Addition on Elliptic Curves

This first procedure tests if (x,y) lies on the curve $y^2=x^3+ax+b$ mod n. The point at infinity is considered a legitimate point.

```
testEC := (x,y,a,b,n) -> if (x=infinity and y=infinity)
or _mod(x^3+a*x+b-y^2, n) = 0
then TRUE else FALSE end_if:
```

A few tests:

```
testEC(3,6,1,6,5782975329)
```

```
 TRUE
```

```
testEC(infinity,infinity,1,6,5782975329)
```

```
 TRUE
```

The next procedure computes elliptic curve addition mod n.

```
addEC := proc(x1,y1,x2,y2,a,b,n)
local z, m, num, den, x3;
begin
if (x1=infinity and y1=infinity)
   then return([x2,y2]) end_if;
if (x2=infinity and y2=infinity)
   then return([x1,y1]) end_if;
if (x1=x2 and y1=y2 and y1=0)
   then return([infinity,infinity]) end_if;
if (x1=x2 and y1<>y2)
   then return([infinity,infinity]) end_if;
if (x1=x2 and y1=y2) then den:=2*y1 else
   den:=x2-x1 end_if:
z := gcd(den,n):
if (z<>1 and z<>n) then print("found factor of n", z);
   return() end_if:
if (x1=x2 and y1=y2) then num:=3*x1^2+a else
   num:=y2-y1 end_if:
m := powermod(den, -1, n)*num:
x3 := _mod(m^2 - x1 -x2, n):
return([x3, _mod(m*(x1-x3)-y1, n)])
end_proc:
```

The next procedure multiplies a point on an elliptic curve mod n by a given positive integer.

```
multEC := proc(x,y,mult,a,b,n)
local z, out, x1, y1;
begin
z := mult: out:= [infinity,infinity]: x1:=x: y1:=y:
while z<>0 do
   while _mod(z,2)=0 do z:=z/2:
   [x1,y1] := addEC(x1,y1,x1,y1,a,b,n):
   end_while:
   z:=z-1:
   out := addEC(x1,y1,out[1],out[2],a,b,n):
end_while;
return(out)
end_proc:
```

A few tests of how this works:

```
[x,y]:=addEC(3,6,3,6,1,6,5782975327)
```

```
 [3212764070, 2141842716]
```

```
[x,y]:=multEC(3,6,2,1,6,5782975327)
```

```
[x,y]:=multEC(3,6,2,1,6,5782975327)
```
```
[3212764070, 2141842716]
```
```
[x,y]:=multEC(3,6,3,1,6,5782975327)
```
```
[813230904, 5658731715]
```
```
addEC(3,6,3212764070, 2141842716,1,6,5782975327)
```
```
[813230904, 5658731715]
```

Now let's use this for factoring products of 5-digit primes.

```
r:=random(10^5): n:=nextprime(r())*nextprime(r())
```
```
674637049
```

Let's try factoring with ifactor:

```
ifactor(n)
```
```
22091 30539
```

Now let's try with the elliptic curve method.

```
[x,y]:=multEC(3,6,fact(100),1,6,n)
```
```
"found factor of n", 30539
Error: Illegal operand [_index];
during evaluation of 'multEC'
```

Or with another curve:

```
[x,y]:=multEC(1,1,fact(100),5,-5,n)
```
```
[151480022, 157907040]
```

Or still another curve:

```
[x,y]:=multEC(1,2,fact(100),7,-4,n)
```
```
"found factor of n", 22091
Error: Illegal operand [_index];
during evaluation of 'multEC'
```

```
```