The University of Sydney
Math3925 Public Key Cryptography

Semester 2 | **Exercises for Week 4** | 2004

In order to implement an index calculus algorithm, we need a smoothness algorithm:

```
function IsSmooth(m,prms)
   // Returns true if and only if m factors over the prime
   // sequence prms, and if so, returns the exponent vector.
   error if m eq 0, "Argument 1 must be nonzero.";
   v := Vector([ 0 : i in [1..#prms] ]);
   for k in [1..#prms] do
       p := prms[k];
       if p eq -1 then
           if m lt 0 then
               v[k] +:= 1; m *:= -1;
           end if;
       else
           while m mod p eq 0 do
               v[k] +:= 1; m div:= p;
           end while;
       end if;
   end for;
   if m ne 1 then return false, _; end if;
   return true, v;
end function;
```

A smoothness base of t elements can be generated with a simple function:

```
function SmoothnessBase(t)
    prms := [ -1 ];
    p := 2;
    for i in [2..t] do
        Append(~prms,p);
        p := NextPrime(p);
    end for;
    return prms;
end function;
```

With these two functions, we can search for relations in the multiplication group $\mathbb{Z}/n\mathbb{Z}^*$. A simple index calculus algorithm is realised in the following lines of code:

```
function ModularRelations(n,prms,b,t)
    Z := Integers();
    R := ResidueClassRing(n);
    rels := [ RSpace(Z,#prms) | ];
```

```
    for k in [1..t] do
        u := Vector([ Random([0..b]) : i in [1..#prms] ]);
        m := Z!&*[ R!prms[i]^u[i] : i in [1..#prms] ];
        bool, v := IsSmooth(m,prms);
        if bool then
            Append(~rels,u-v);
        end if;
    end for;
    return rels;
end function;
```

1. **a.** Use the above functions to determine a set of prime generators and the complete sets of relations among them in $\mathbb{Z}/n\mathbb{Z}^*$ for $n = 2^{29} - 1$.

b. Use the relations to realise a factorization of n.

c. How does this method compare to a Pollard ρ factorization?

2. **a.** Similarly find a set of generators and relations for the group $\mathbb{Z}/p\mathbb{Z}^*$ for the prime $p = 2^{31} - 1$.

b. Solve the discrete logarithm $\log_3(5)$ in this group using these relations.