MATH 467, THE QUADRATIC SIEVE (QS)

Algorithm QS. We are given an odd number n which we know to be composite and not a perfect power. The objective is to find a non–trivial factor of n by first finding x and y so that $x^2 \equiv y^2 \pmod{n}$ and then checking $GCD(x \pm y, n)$.

A number $m \in \mathbb{N}$ is called *B–factorable* when it has no prime factor exceeding B.

1. *Initialization.*

1.1. Pick a number B for the size of the factor base. Theory says take $B = \lceil L(n)^{1/2} \rceil$ where $L(n) = \exp(\sqrt{\log n \log\log n})$, but in practice a B somewhat smaller works well. Also, adding extra primes suggested by the sieving process can be useful and if one uses the wrinkle in 5.3 the prime p is adjoined to the factor base.

1.2. Set $p_0 = -1$, $p_1 = 2$ and find the odd primes $p_2 < p_3 < \ldots < p_K \le B$ such that $\left(\frac{n}{p_k}\right)_L = 1$. Here $K+1$ is the cardinality of the factor base. Algorithm LJ is useful here (described elsewhere).

1.3. For $k = 2, \ldots, K$ find the solutions $\pm t_k$ to $x^2 \equiv n \pmod{p_k}$ by using algorithms QC357/8 and QC1/8 (described elsewhere).

2. *Sieving.*

2.1. Let $N = \lceil \sqrt{n} \rceil$. For each $x = N + j$, $j = 0, \pm \ldots$ the $x^2 - n$ will be sieved until one has obtained a list of at least $K+2$ B-factorable $x^2 - n$ and their factorizations. This could be done by using a matrix, with B^2 columns (B^2 is somewhat arbitrary and can be increased if necessary) so that each column is a $K+3$ dimensional vector in which the first entry is x, the second is $x^2 - n$, and the $k+3$–rd entry will be the exponent of p_k in $x^2 - n$.

2.2. For each prime p_k in the factor base divide out all the prime factors p_k in each entry $x^2 - n$ with $x \equiv \pm t_k \pmod{p_k}$, recording the exponent in the $k+3$-rd entry in the associated j-th vector.

2.3. If the second entry in the j–th vector has reduced to 1, then $x^2 - n$ is B–factorable. Relatively few will be completely factored. Discard those x which don't completely factorise in the factor base. Theory tells us that we will need at least $K+1$, and generally somewhat more, say J, completely factored, which is the reason for taking so many columns in the first place. In my model solutions I take $J = K + 9$ but this is probably overkill.

3. *Linear Algebra.*

3.1. Form a $(K+1) \times J$ matrix $\mathcal{M}$ with the columns being formed by the 3–rd through $K+3$–rd entries of the column vectors arising in 2.2 from the B-factorable $x^2 - n$, but with the entries reduced modulo 2. It is convenient to label columns as $j = 1$ through J and the corresponding x as x_1 through x_J.

3.2. Use linear algebra (e.g. Gaussian elimination) to solve $\mathcal{M}\mathbf{e} = \mathbf{0} \pmod{2}$ where $\mathbf{e} = (e_1, e_2, \ldots, e_J)$ is a J dimensional vector of 0s and 1s (not all 0!). It is likely that one will need more than one solution before finding a factorization of n. Gaussian elimination or standard linear algebra packages should give a basis for the space of all solutions.

4. *Factorization.*

4.1. Compute $x = x_1^{e_1} x_2^{e_2} \ldots x_J^{e_J}$ modulo n and

$$y = \sqrt{(x_1^2 - n)^{e_1}(x_2^2 - n)^{e_2} \ldots (x_J^2 - n)^{e_J}} \pmod{n}$$

modulo n. The value of x can be computed by using the first entries in the column vectors in the original matrix and the square root in the definition of y should be computed using the factorizations in the body of that matrix. Note that all multiplications should be performed modulo n so nothing bigger than n^2 will occur.

4.2. Compute $l =$GCD$(x - y, n)$, $m =$GCD$(x + y, n)$.

4.3. Return l, m.

5. *Aftermath.*

The method described above should work for the examples in the final project. In more difficult cases the following has been tried.

5.1. If none of the l, m are proper factors of n try one or more of the following.

5.2. Extend the sieving in 2.1 to obtain more x_j and so more pairs.

5.3. Use another polynomial in place of $x^2 - n$, or rather, be a bit more cunning about the choice of the x in 2.1. Choose a large prime p for which $b^2 - n \equiv 0 \pmod{p}$ is soluble, and compute b. Then $(px+b)^2 - n \equiv 0 \pmod{p}$ and x can be chosen so that $f(x) = ((px+b)^2 - n)/p$ is comparatively small since p is large, so the sieving proceeds relatively speedily, there is a better chance of a complete factorization of $f(x)$, and we only have to augment the factor base with the prime p.