Computation elements needed to implement RSA

In order for RSA to be a useful system to implement, it must be possible to do the following types of calculations relatively quickly for very large integers. The complexity of these operations should grow no faster than $O((\text{number of digits})^d)$ in order for this system to be practical.

- compute $\phi(n)$ given $n = p_1^{a_1} p_2^{a_2} \cdots p_k^{a_k}$
- calculate a^b modulo n.
- determine the inverse of an integer a modulo n.
- find very large primes

The security of the RSA system relies on the fact that given an integer n which is the product of two large primes, it is computationally difficult to factor n. Therefore if we are to play the role of the opponent we would like an algorithm which also runs relatively quickly which can:

- factor very large integers