9. (a) Using the big-O notation, estimate in terms of a simple function of n the number of bit operations required to compute 3^n in binary.
(b) Do the same for n^n.
10. Estimate in terms of a simple function of n and N the number of bit operations required to compute N^n.
11. The following formula holds for the sum of the first n perfect squares:

$$\sum_{j=1}^{n} j^2 = n(n+1)(2n+1)/6.$$

(a) Using the big-O notation, estimate (in terms of n) the number of bit operations required to perform the computations in the left side of this equality.
(b) Estimate the number of bit operations required to perform the computations on the right in this equality.
12. Using the big-0 notation, estimate the number of bit operations required to multiply an $r \times n$-matrix by an $n \times s$-matrix, where all matrix entries are $\leq m$.
13. The object of this exercise is to estimate as a function of n the number of bit operations required to compute the product of all prime numbers less than n. Here we suppose that we have already compiled an extremely long list containing all primes up to n.
(a) According to the Prime Number Theorem, the number of primes less than or equal to n (this is denoted $\pi(n)$) is asymptotic to $n/\log n$. This means that the following limit approaches 1 as $n \longrightarrow \infty$: $\lim \frac{\pi(n)}{n/\log n}$. Using the Prime Number Theorem, estimate the number of binary digits in the product of all primes less than n.
(b) Find a bound for the number of bit operations in one of the multiplications that's required in the computation of this product.
(c) Estimate the number of bit operations required to compute the product of all prime numbers less than n.
14. (a) Suppose you want to test if a large odd number n is a prime by trial division by all odd numbers $\leq \sqrt{n}$. Estimate the number of bit operations this will take.
(b) In part (a), suppose you have a list of prime numbers up to $\sqrt{n}$, and you test primality by trial division by those primes (i.e., no longer running through all odd numbers). Give a time estimate in this case. Use the Prime Number Theorem.
15. Estimate the time required to test if n is divisible by a prime $\leq m$. Suppose that you have a list of all primes $\leq m$, and again use the Prime Number Theorem.
16. Let n be a very large integer written in binary. Find a simple algorithm that computes $[\sqrt{n}]$ in $O(\log^3 n)$ bit operations (here $[\ ]$ denotes the greatest integer function).