is a function that returns the value of the square root of sart(x) x. Thus, sqrt(9) becomes 3, sqrt(169) becomes 13, sqrt(10.24) becomes 3.2, and sqrt(0.5625) becomes .75. Note that it is invalid to ask the sqrt(x) function to find the square root of a negative value. That is, the domain of the sqrt(x) function is the non-negative real numbers (or in some cases their approximations). Also, note that the answer produced will be a very good answer, but it is not always the correct answer. For example, in perl, one popular computer language, sqrt(2) returns the value 1.4142135623731. This is a great answer, but the square root of 2 is an irrational number and as such it is a nonterminating, non-repeating decimal. Therefore, the answer 1.4142135623731 cannot be correct. The Javascript version of sqrt(x), called as Math.sqrt(2), produces 1.4142135623730951, a better answer, but still not the complete answer.

log(x) is the logarithmic function. In some, but not all, languages log(x) refers to the base ten logarithm, called the common logarithm. That is, log(100) becomes 2 because  $10^2$  is 100. Similarly, log(100000) is 5 because  $10^5$  is 100000, and log(1) is 0 because  $10^0=1$ . We can continue this with log(1/1000) is -3 because  $10^{-3}$  is 1/1000. Values that are not so easily computed are, for the most part, irrational numbers and therefore the result of the log(x) function is often just a great approximation to the exact answer. Thus, log(2) yields a value such as 0.30102999566398 on a TI-83 calculator. This is a good approximation because 10<sup>0.30102999566398</sup> is approximately 2. In virtually all of mathematics we use log(x) to mean the log base 10 of x, just as we have seen above. Imagine your shock, then, when both perl and Javascript produce the value 0.693147180559945 for log(2). That different answer is because in those languages log(x) does not mean the log base 10 of x. In those languages, log(x) means the log base e, where e is the irrational number that is approximately 2.718281828459. It is because 2.718281828459<sup>0.693147180559945</sup> is approximately 2 that those languages return the value 0.693147180559945 for the function log(2). In mathematics, when we want the logarithm base e we refer to it as ln(x), the natural logarithm function. It is important to note that the domain of the log(x) function, whether it is the common logarithm or the natural logarithm, is the set of positive real numbers. We cannot raise 10 or *e* to a power and get the value 0. Nor can we raise either to a power and get a negative value.

In(x) is the natural logarithm function. Thus, as we saw in the discussion of the log(x) function, to say that the ln(x) is y means that  $e^{y}$  is equal to x, where e is that special number that is approximately 2.718281828459. The ln(x) function is well recognized in mathematics, but, again as we have seen, it is sometimes given a different name, unfortunately log(x), in some computer languages. In many science and math problems, and even in some economics problems, we have need of the natural

logarithm. If you are solving the problem by hand then you will certainly want to use ln(x) to express the natural logarithm. If, on the other hand, you are writing a program to solve or work with the problem, then you will need to see just how your computer language of choice represents the natural logarithm.

exp(x) is a function to produce the value of  $e^x$ . This is the inverse function for the natural logarithm function. The domain for this function is any real number because we can raise e to any power. However, the range of the function is only the positive real numbers because e raised to any real power will not be 0 or less than 0.