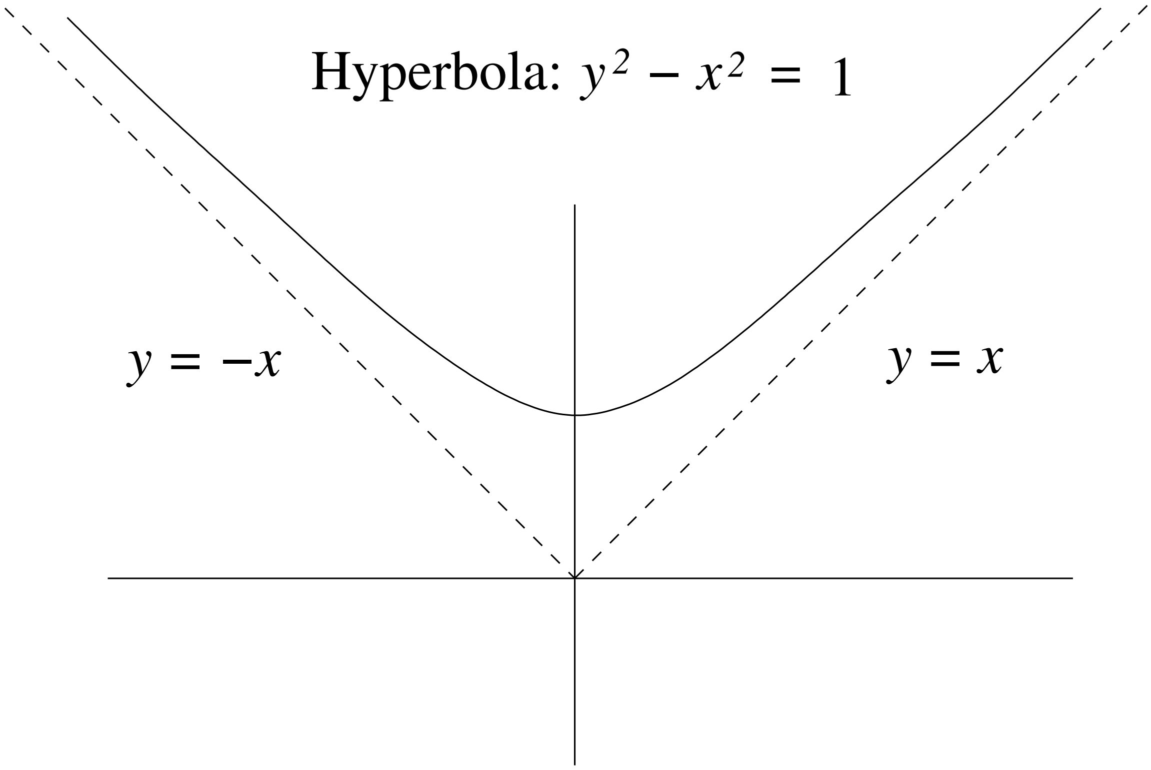
Running time

Therefore, the motivations for a study on running time of algorithms are many, though the most important being is scaling and performance guarantees. Other important aspects of study on algorithmic efficiency:

- the generic characteristics of growth functions for expressing the running time of algorithms,
- the generic mathematical tools, and a framework for comparison of relative performances of running time of alternative algorithms.

Asymptotic functions

The word *asymptotic* means *not falling together*. The idea of asymptotic came from study of hyperbolas. The two extending ends such as $y^2 - x^2 = 1$ shown below while quite close to lines $y = x$ and $y = -x$ do not touch.



Time complexity

- In case of the time complexity analysis, asymtotics, is used to mean *almost falling together*.
- In other words, the interpretation of *affinity* or *closeness* inherent with the original meaning of *not falling together* for asymptotics is meant in analyzing the closeness of running times of different algorithms.
- Certain specific asymptotics that allow us to compare the growth rates of the functions.

Big Oh notation

- Introduced by Bachmann in 1894.
- It filters out unnecessary details and allows to concentrate on the dominating term.
- For large problem size, the imprecision of the kind big Oh notation, denoted by "O", is of no concern.
- At the same time the details captured by O notation is just appropriate to compare the relative growth rate of functions.

Big Oh

- See the following example

$$T(n) = \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n$$

- The dominant term in the function $T(n)$ is $\frac{1}{3}n^3$.
- Suppose we write a JAVA program, given in next transparency and compute values of $T(n)$, $\frac{1}{3}n^3$, and the relative percentage in differences in the two computed values.