

Course Objectives and Outcomes

Course Objectives

This course introduces students to methods used to design and analyze algorithms.

Upon completion of 6.046, students will be able to analyze worst-case running time of algorithms and understand fundamental algorithmic problems. They will know and be able to apply basic paradigms and data structures used to solve algorithmic problems.

Course Outcomes

Students will be able to:

1. Compare asymptotic behavior of functions obtained by elementary composition of polynomials, exponentials and logarithmic functions.
2. Solve systems of recurrence relations, especially those that arise commonly in the analysis of algorithms.
3. Analyze worst-case running time of programs using:
 - (a) Linear recurrences to model running time of recursive programs.
 - (b) Amortized analysis, and the method of potential functions.
4. Apply fundamental algorithmic *paradigms* in design of algorithms. Specific examples include:
 - (a) The Divide and conquer paradigm.
 - (b) Dynamic programming.
 - (c) Greedy algorithms (and when they work).
5. Apply basic *data structures* and analyze their performance. Specific examples include heaps, queues (priority queues), search trees and hash functions.
6. Understand and apply fundamental algorithmic *problems* including:
 - (a) Sorting and Searching
 - (b) Graph traversal and shortest paths
 - (c) Network flows
7. Be able to prove inherent limitations to efficiency, due to:
 - (a) Information-theoretic lower bounds.
 - (b) NP-completeness.
8. Understand the motivation for algorithmic problems as brought out by engineering and scientific applications.