# logo-medWorkshop #2: Decision Trees

Today we used decision trees in two different kinds of problems. In NIM, we could model the game as a series of actions by each player. Ultimately, a set of series of actions would eventually lead to a win or a loss. By drawing a decision tree and following it, one of the players could ensure that she won. Furthermore, by assuming that your opponent plays an optimal strategy, you could also determine an optimal strategy and thereby write an algorithm that that would always win.

NIM is a very simple game, but similar kinds of modeling is used when computers play two-player games like checkers, chess or Go. In NIM, there were only three possible actions, or “branches”, at each turn. In chess, there are on average 35 possible actions per turn. This makes the decision tree huge. Using *artificial intelligence algorithms*, computers have successfully beaten world champions in chess (like Kasparov, to the right). The next challenge for computer-playing games is the Chinese game Go: in Go, on average, there are 361 legal moves per turn!

For the coin-balancing problem, we modeled the solution algorithm as a decision tree. Each point in the tree was a weighing, and each branch represented a different outcome of the weighing (balanced, left side heavy, right side heavy). This kind of decision tree is also used in *operations research* (also called Industrial Engineering, or IEOR at Columbia) to model complex decision making. It is also closely related to a CS field called *information theory*. At each point in the tree, we learn more information, and get closer to a solution.

### Related Columbia Classes

Core classes:

* COMS W3137 or W3139 Data Structures and Algorithms
* COMS W3203 Discrete Mathematics

Advanced classes in COMS and IEOR:

* COMS W4231 Analysis of Algorithms, I
* COMS W4701 Artificial Intelligence
* COMS E6717 Information Theory
* IEOR E4003 Industrial Economics
* IEOR E4201x The Engineering of Management, I