This is a summary of the essential aspects of the extensive form of a game of complete information. This form is a particularly convenient way of modeling games in which players act sequentially, i.e., one player chooses an action and then another player, who knows the first player's choice (complete information), chooses an action and so forth.

The extensive form is characterized by a set of players and a tree (directed graph) with nodes, branches and leaves. Each node has a label naming the player who gets to “play” if the node is reached and each departing branch has a label naming one of the actions that could be chosen at the node. Departing branches end either at another node or at a (terminal) leaf. A leaf represents an end of the game and has a label with a tuple containing the payoffs of the players listed, by convention, in the order in which the players move. At most one branch can arrive at a node or leaf — the beginning node has none and all the rest have one. There is, therefore, a unique path of branches to any particular node or leaf and this path is called the history of the node or leaf.

Here, for example, is a tree in which nodes appear as ovals, leafs as rectangles and branches as lines. It corresponds to a game in which Player 1 moves first and can choose either A or B. Player 2 moves next and can choose either C or D after the history in which Player 1 chose A and either E or F after the history in which Player 1 chose B. Payoffs are listed in the rectangular leaf nodes, e.g., Player 1’s payoff after the history AC is 4 and Player 2’s payoff is 2.

A strategy for a player is a list of actions, one for each node named for the player. Player 1 has only one node (with null history) and since there are two choices at this node, Player 1 has two strategies, A and B. Player 2 has two nodes, one with history A and another with history B. A strategy for Player 2 must then specify an action for each of these two nodes. We follow the convention of listing the actions in the same order as the nodes appear from left to right so that, for example, the strategy CE corresponds to choosing C after history A and E after history B. Player 2 has four possible strategies: CE, DE, CF, and DF.

To identify the Nash equilibria of this game, it is convenient to construct the strategic form. Note that best replies are marked with †’s and that there are three Nash equilibria in pure strategies: (A, CE), (A, CF) and (B, DF).

Extensive form games typically have subgames. For example, the game
that begins after history $A$ when Player 2 chooses an action is a subgame. Similarly, the game that begins after history $B$ is another subgame. In general, subgames correspond to portions of the tree that satisfy the same requirements as were imposed on the whole tree.

A Nash equilibrium requires a profile of strategies to be mutually best replies in the entire game. We can now require that a profile of strategies be mutually best replies in every subgame as well. In other words, we can require that a profile of strategies be a Nash equilibrium in every subgame as well as in the entire game. Such a profile is called a *subgame-perfect Nash equilibrium* and, since we are imposing addition conditions, the set of subgame-perfect Nash equilibria will typically be a strict subset of the set of Nash equilibria.

How do we find subgame-perfect Nash equilibria? We use a process called *backward induction* that, as the name suggests, involves working backwards through the extensive form. That is, we begin by looking at the bottom nodes — the ones that lead only to terminal leaves.

The node that follows history $A$ is an example of such a “bottom” node. In this subgame, Player 2 is the only player and to be Nash in this subgame requires that Player 2’s choice be optimal. This means that Player 2 must choose $C$ at this node since $2 > 1$. Similarly, at the node following history $B$, Player 2 must choose $F$ since $4 > 3$. $CF$ is, therefore, the only strategy for Player 2 that satisfies the subgame-perfection requirement.

Now that we know that Player 2 must choose $CF$, we can move back up the tree to consider Player 1’s choice. Since combined with $CF$ choosing $A$ gives a payoff of 4 while choosing $B$ gives a payoff of 2, Player 1 must choose $A$ to satisfy the subgame-perfection requirement.

We can summarize our reasoning in the tree diagram by erasing the edges that are inconsistent with subgame-perfection. Note that of the three Nash equilibria, $(A, CE), (A, CF)$ and $(B, DF)$, only $(A, CF)$ is a subgame-perfect Nash equilibrium.

Backward induction is not only easy to do, it is guaranteed to produce a subgame-perfect equilibrium. One caveat: whenever there are ties in which the largest payoff for a player is associated with more than one action, you must make a copy of the tree for every such action with the other actions erased. Multiple subgame-perfect equilibria can only arise through such ties.

Life can only be understood backwards; but it must be lived forwards.

— Soren Kirkegaard