Машина Тьюринга. МТ — это абстрактная модель вычислений, которая состоит из каретки и бесконечной ленты, по которой каретка будет двигаться. Программа для МТ — это автомат, который по состоянию и букве на позиции каретки говорит, что должна сделать каретка и в какое состояние должен перейти алгоритм. То есть, написать что-то в текущую клетку, сдвинуть каретку на позицию вбок. Будем считать, что P — это такой класс задач, для которых ответ бинарен, и для любого ввода алгоритм для МТ работает за полиномиальное время от размера ввода. Соответственно, для них автомат тоже имеет полиномиальный размер.

NP-полнота. Введем несколько задач с бинарным ответом (то есть да/нет), которые мы будем хотеть решать:

- Lin solver — решение системы линейных уравнений (Полиномиальна — алгоритм Гаусса)
- SAT — Задача о булевой разрешимости
- CNF-SAT — SAT, который задается набором конъюнктов, каждый конъюнктов внутри состоит из скольки-то дизъюнктов.
- Suset-sum — задача о рюкзаке
- k-Clique — задача о поиске клики размера k.
- Ham — задача о поиске Гамильтонова цикла/пути.
- Euler — Задача о поиске эйлерова цикла/пути (опять-таки, полиномиальна)

Эти задачи интересны тем, что полиномиального алгоритма решения для них пока что нет (ну кроме тех, про которые я написал обратное), но зато есть полиномиальный способ проверить, верен ли положительный ответ (сертификат). Для P-задач алгоритм проверки совпадает с решением — само условие задачи уже является своим сертификатом.

Будем считать, что задачи NP (non-deterministic polynomial) — это такие задачи, которые разрешимы на недетерминированной машине Тьюринга. В недетерминированной машине Тьюринга надо делать одну из операций, записанных в ячейке. Раньше в ячейке была только одна команда, а теперь может быть несколько — мы сами можем выбрать, какое нас интересует. Задача разрешима на такой машине, если все пути в дереве разбора имеют полиномиальный размер. Например, в задаче Subset-sum можно сделать автомат вида «полное бинарное дерево», глубина которого будет полиномиальна, а листов будет 2^n. Тогда сертификатом будет просто путь в этом дереве.

Если у задачи есть возможность полиномиально проверить отрицательный ответ, то она принадлежит классу $co - NP$.

Задача называется NP-трудной, если она хотя бы так же трудна, как и любая другая задача из NP.

Что такое «так же трудна»? Задача A сводится к задаче B, мы должны предъявить полиномиальный алгоритм, который преобразует задачу A в задачу B (переводит вход в вход B, решение B в решение A). То есть, задача называется NP-трудной, если ее полиномиальное решение автоматически решит все задачи из NP.

Задача называется NP-полной, если она и NP и NP-трудная.

Теорема Кука. Любая задача из NP может быть записана в виде CNF-SAT. По любой задаче из NP мы знаем ее дерево решения для недетерминированной машины Тьюринга. Кроме того, у нас есть детерменированная МТ, которая умеет проверять сертификат. Рассмотрим эту МТ. Мы можем взять все состояния во все моменты времени как переменные, все положения каретки во все моменты времени как переменные, все символы на ленте во все моменты времени как переменные. Тогда в каждый момент времени для нас эти параметры задают булевый вектор.

- В булевом векторе для состояния, положения и символов одной позиции должно быть ровно по одной единице.
- Значения для всех символов в момент времени 0 должны совпадать со стартовыми значениями, кроме тех позиций, которые соответствуют сертификату — они могут быть любыми.
- Поскольку у нас везде по одному переходу, то нам надо построить граф имликаций.
- Кроме того, надо сделать условие, что символ меняется только под кареткой. Это тоже какие-то импликации

Схема будет иметь полиномиальный размер. Решение схемы явно задаст стартовые переменные, которые выдадут сертификат-решение, поэтому решение схемы решит и задачу.

В целом, подробное доказательство гораздо сложнее, но тут уже можно додумать, а громоздких записей нет.