How to Prove Membership in P or NP

To prove that a given language is in P:

- Construct an algorithm that decides the language.
- This algorithm may "call" any other algorithms from the textbook, lectures, class handouts, or homework assignments (but you should cite the appropriate reference).
- How do you decide which existing algorithms it should call, if any? This is a familiar problem for any program designer. You may be expected to figure this out for yourself or a hint may be provided. When proving closure of P under a given operation the obvious choice is an assumed polytime decider for a given member of P.
- Prove that the language it recognizes is equal to the given language and that the algorithm runs in polynomial time.

To prove that a given language is in NP:

Either:

- Construct (a high-level description of) an NTM (equivalently, a nondeterministic algorithm) that decides the language.
- This nondeterministic algorithm may "call" any other algorithms from the textbook, lectures, class handouts, or homework assignments (but you should cite the appropriate reference).
- How do you decide which existing algorithms it should call, if any? This is a familiar problem for any program designer. You may be expected to figure this out for yourself or a hint may be provided. When proving closure of NP under a given operation the obvious choice is an assumed nondeterministic polytime decider for a given member of NP.
- Prove that the language it recognizes is equal to the given language and that the algorithm runs in nondeterministic polynomial time.

Or:

- Specify a certificate that can be used with a verifier to decide the language.
- Give a verifier that uses that certificate to verify membership in the given language.
- Prove that the language recognized by the verifier is the given language and that the verifier runs in polynomial time.

In all the cases we study, the certificate/verifier approach is so obvious that it generally takes about one or two sentences to describe the certificate and about one or two additional sentences to describe how the verifier works.