

# Make A Plan - How to Solve Problems

- There is a simple rule: do not just dive in to coding! Repeat after me: DO NOT JUST DIVE INTO CODING.
- According to data we have gathered internally from the results of mass interview practice, the failure mode of entry-level programmers is not making enough of a plan before they began implementation.

## Really read the problem

- Read it slowly.
- Read it again.
- Think to yourself, "What assumptions are hidden in this problem?"
- Identify input and outputs.

## Outline your approach

- Decide on your core data structure or data type.
- What are the start and end states of that data?
- How does that data "mutate" from start state to end state?
  - Outline your state-changing logic in English pseudo-code.
  - Re-read it, edit it, make it clear and precise.

- Outline your structure with one or more empty functions (aka stub functions).
- Be sure to give each stub a clear name.

Implement carefully, step-by-step.

- Fill in the main function body.
- Stub out helper function (if desired)
- Each function should ideally have a single purpose. (This will not always be the case during your interview/admissions assessment) The name of the function should reflect clearly what it will do.
- Test as you go.