# Problem Solving

- [ ] 1.1 (Understanding the problem): Can I restate the problem in my own words?
- [ ] 1.2 (Understanding the problem): What are the inputs that go into the problem?
- [ ] 1.3 (Understanding the problem): What are the outputs that should come from the solution to the problem?
- [ ] 1.4 (Understanding the problem): Can the outputs be determined from the inputs? In other words, do I have enough information to solve the problem?
- [ ] 1.5 (Understanding the problem): How should I label the important pieces of data that are a part of the problem?
- [ ] 2.1 (Explore Concrete Examples): Start with Simple Examples
- [ ] 2.2 (Explore Concrete Examples):Progress to More Complex Examples
- [ ] 2.3 (Explore Concrete Examples): Explore Examples with Empty Inputs
- [ ] 2.4 (Explore Concrete Examples): Explore Examples with Invalid Inputs
- [ ] 3.1 (Break It Down): Explicitly write out the steps you need to take
- [ ] 4.1 (Solve/Simplify): Solve the problem, If you can't, solve a simpler problem!
- [ ] 4.2 (Solve/Simplify): Find the core difficulty in what you're trying to do
- [ ] 4.3 (Solve/Simplify): Temporarily ignore that difficulty
- [ ] 4.4 (Solve/Simplify): Write a simplified solution
- [ ] 4.5 (Solve/Simplify): Then incorporate that difficulty back in
- [ ] 5.1 (Look Back and Refactor): Can you check the result?
- [ ] 5.2 (Look Back and Refactor): Can you derive the result differently?
- [ ] 5.3 (Look Back and Refactor): Can you understand it at a glance?
- [ ] 5.4 (Look Back and Refactor): Can you use the result or method for some other problem?
- [ ] 5.5 (Look Back and Refactor): Can you improve the performance of your solution?
- [ ] 5.6 (Look Back and Refactor): Can you think of other ways to refactor?
- [ ] 5.7 (Look Back and Refactor): How have other people solved this problem?

**Make and Share Free Checklists**
checkli.com