# How to Solve It: Key Problem-Solving Questions

Your ability to solve problems is infinite. This is not something you are born with. Instead, you can grow your problem-solving skills by working hard and working smart. In this post, I share simple problem-solving strategies that you can use to make progress on any problem you face. If you leverage these strategies and repeat these practices over long periods of time, you will transform yourself into an expert problem solver.

Below is a list of over 60 questions to get your problem-solving juices flowing. I organize these questions into four different phases of problem solving. To generate this list, I relied heavily on what I call the five-volume bible of problem solving:

1. *How to Solve It: A New Aspect of Mathematical Method*
2. *Mathematics and Plausible Reasoning, Volume 1: Induction and Analogy in Mathematics*
3. *Mathematics and Plausible Reasoning, Volume 2: Patterns of Plausible* Inference
4. *Mathematical Discovery: On Understanding, Learning, and Teaching Problem Solving, Volume 1*
5. *Mathematical Discovery: On Understanding, Learning, and Teaching Problem Solving, Volume 2*

All five of these books were written by George Pulya. In this introductory post, we focus on strategies from the four-step process to solve problem outlined in *How to Solve It*.

I encourage you to use this list as a check list. As you embark on each problem-solving adventure, choose a question from the appropriate phase of problem solving. Then, write as much as you can in response to this item. After you've exhausted your mind on a specific item, move to some other question from that phase. Slowly and methodically work your way through these questions. Do your best to practice leveraging all the questions in all the stages. This doesn't mean you have to practice all the questions on a single problem. But, as you continue to solve problems, make sure that you are leveraging all the questions on this list. There are great lessons and unique insights that come from meditating on each question. Together, these form the foundation of a problem-solving habit that can catapult you to being able to solve almost any problem you can imagine.

## STEP 1: UNDERSTAND THE PROBLEM

- ☐ What is *unknown*?
- ☐ What are the data?
- ☐ What is the condition?
  - ☐ Is it possible to satisfy the condition?
  - ☐ Is the condition sufficient to determine the unknown?
  - ☐ Separate the various parts of the condition.

- ☐ What types of example(s) can you generate to learn more about the problem?

- ☐ How can you visualize this problem?
  - ☐ What pictures or diagrams can you draw to illuminate key features?
  - ☐ Draw at least one diagram to describe the problem.
  - ☐ Label the diagram(s) with key variables.

- ☐ What notation or symbol(s) makes sense to use for this problem?
  - ☐ Why do you like that notation?
  - ☐ Define each variable verbally.

- ☐ What questions come up for you as you meditate on this problem?
- ☐ Describe the problem in abuelita (simple, intuitive) language
- ☐ Describe the problem in nerdy (formal, technical) language

## STEP 2: DEVISE A PLAN

- ☐ What problem(s) might be related to this one?
  - ☐ Have you seen this problem before?
  - ☐ Have you seen this problem in a slightly different form?
  - ☐ What related problems have you solved before?
  - ☐ How can you use your previous results, solutions, or methods on those related problems here?
  - ☐ Look at the unknown(s). What problem(s) do you know that have the same/similar unknowns?
  - ☐ What auxiliary elements can you introduce to allow you to use your work solving other problems on this problem?

- ☐ What theorem(s) could be useful?
  - ☐ Write the theorem statement(s) out in full.
  - ☐ Translate the notation of the problem into the notation of the theorem or vice versa.

- ☐ What definition(s) do you know that could be useful?
  - ☐ Write the definition statement(s) out in full.
  - ☐ If so, write the definition(s) out in full. Make sure to explicitly connect the notation of the problem to the notation in the definition.

If you can't solve the proposed problem, create and solve any related problem(s):

- ☐ How can you simplify the problem?
  - ☐ What is a more accessible, related problem?
  - ☐ What is a more general problem?
  - ☐ What is an analogous problem?

- ☐ Identify different parts of the problem?
  - ☐ What can you do to solve the parts of the problem that you find?
  - ☐ What if you keep only a part of the condition and drop the other part(s)?
  - ☐ Identify subproblems within the larger problem
  - ☐ What can you do to solve the subproblems you find?

- ☐ What can you learn from the unknown or the given data?
  - ☐ If you drop some of the conditions, how does that effect the unknown(s) in the problem?
  - ☐ What patterns, insights, or questions can you create from the data in the problem?
  - ☐ What other data might be useful for determining the unknown(s)?
  - ☐ How can you change the unknown or the data or both to produce new unknowns or data that are nearer to each other?

- ☐ Have you used all the information you can?
  - ☐ Have you used all the data?
  - ☐ Have you used the entire condition?
  - ☐ Have you taken into account all essential notations involved in the problem?

- ☐ What types of example(s) can you generate to give you insights into how to solve the problem?

## STEP 3: CARRY OUT THE PLAN

- ☐ Carry out your plan of the solution.
- ☐ Can you see clearly that the step(s) are correct?
- ☐ How might you prove that the step(s) are correct?
- ☐ What types of example(s) can you generate to check that your results work?

## STEP 4: LOOK BACK

- ☐ How can you check the results?
- ☐ How can you check the argument?
- ☐ How might you derive the results differently?
- ☐ Can you see the result(s) at a glance?
- ☐ How might you use the result or method for some other problems?
- ☐ How can you make sense of your results?
- ☐ How would you teach your result to someone who has no knowledge of this problem?
- ☐ Describe your work in abuelita (simple, highly intuitive) language?
- ☐ Describe your work in nerdy (formal, technical) language?
- ☐ How can you revise your notation to improve understanding for others?