

The Four-step Problem-solving Process



George Polya described the experience of problem solving in his book, *How to Solve It*, p. v:

A great discovery solves a great problem but there is a grain of discovery in the solution of any problem. Your problem may be modest; but if it challenges your curiosity and brings into play your inventive facilities, and if you solve it by your own means, you may experience the tension and enjoy the triumph of discovery.

As part of his work on problem solving, Polya developed a four-step problem-solving process similar to the following:

- **Understanding the Problem**

1. Can you state the problem in your own words?
2. What are you trying to find or do?
3. What are the unknowns?
4. What information do you obtain from the problem?
5. What information, if any, is missing or not needed?

- **Devising a Plan**

The following list of strategies, although not exhaustive, is very useful:

1. Look for a pattern.
2. Examine related problems and determine if the same technique can be applied.
3. Examine a simpler or special case of the problem to gain insight into the solution of the original problem.
4. Make a table.
5. Make a diagram.
6. Write an equation.
7. Use a guess and check.
8. Work backward.
9. Identify a subgoal.

- **Carrying out the Plan**

1. Implement the strategy in Step 2 and perform any necessary actions or computations.
2. Check each step of the plan as you proceed. This may be intuitive checking or a formal proof of each step.
3. Keep an accurate record of your work.

- **Looking Back**

1. Check the results in the original problem. In some cases, this will require a proof.
2. Interpret the solution in terms of the original problem. Does your answer make sense? Is it reasonable?
3. Determine whether there is another method of finding the solution.
4. If possible, determine other related or more general problems for which the techniques will work.

These and other general mathematics problem-solving strategies, or rules of thumb for successful problem solving, are called *heuristics*.