

# Chapter 1

## Basics of Programming

Programming, more than anything else, is about giving instructions. Programming is the source of control over microprocessors that are the core of embedded systems.

Before exploring the syntax of any programming language, we are going to explore the basic concepts that will be used to compose the code for a program.

We will explore the concepts of pseudo code, branching, looping, procedures, and how these can be expressed in flowcharts, or comments to facilitate problem solving through programming.

Finally, we will explore some concepts like machine language, low-level programming languages, and high-level programming languages.

### 1.1 Where to begin?

We are going to start with a really simple concept. Something you've probably already explored in a previous class.

How do you make a Peanut Butter and Jelly sandwich?

A first draft of an answer to this question might go like:

1. Apply peanut butter to one slice of bread.
2. Apply jelly to a second slice of bread.
3. Lay second piece of bread on first.

For many, in the United States, these instructions may be adequate enough. However, these instructions make very broad assumptions about the readers knowledge, and established ways of doing things. To provide useful instructions to a broader range of reader, you have to make fewer assumptions, and be more explicit with your instructions.

1. Acquire two pieces of bread with the following dimensions: 10cm wide, 10cm tall, and 1cm thick.
2. Apply 15g of peanut butter to one side of the first slice of bread.
3. Apply 7g of grape preserves to one side of the second slice of bread.
4. Lay the first slice of bread flat, with the peanut butter laying up.
5. Lay the second slice of bread on top of the first, with the jelly side down, so that the peanut butter and jelly are in contact.

With this second draft, the instructions are more explicit, but there are still many assumptions made. For example, does everyone know what peanut butter is? The answer is no. So, again, to provide useful instructions, you have to be still more explicit.

This process can seem tedious, but it is at the root of making computers do what you want them to do. Take a moment to reflect on just how difficult it would be to explain to a 2 year old the process of making a peanut butter and jelly sandwich. The explanation would require details for everything from the process of growing peanuts, making peanut butter, baking bread, preparing jellies, to slicing bread, and the proper tools for applying the core ingredients. Now, take a second to reflect on just how difficult it is to explain the same process to a rock.

At the core of our modern technology is a whole bunch of different configurations of digital circuits etched into silicon crystals. These rocks don't do anything more than what they are designed to do. Things like microprocessors, while incredibly useful for so many tasks, are less useful than a battery unless provided with a program that brings their design to use.

So, to answer the initial question of where to begin, when it comes to programming, the place to begin is at stating the objective. What problem are you trying to solve, and what is the solution to that problem. All you need, at the beginning, is this information in broad terms.

## 1.2 Defining the solution.

Once you have a problem and solution defined in broad terms, it is always a good first step to question the accuracy of both. Is the problem you've identified the real problem? Is the solution you've identified adequate?

Next, it becomes necessary to start breaking the solution down into constituent steps. This first round of definition usually involves drawing connections between the parts of the solution, and the end product.

For example, when talking about making a PB&J, we'd be in a good position if we identified the components that go into making that sandwich, and organizing them that conveys relationships.

- Peanut Butter
- Jelly
- Bread
- Knife
- Plate

**Figure 1.2.1** Illustration of solution for making a Peanut Butter and Jelly Sandwich.

Then