

# What Drives Research into New Programming Languages?

Why isn't C or C++ or C+++ enough?

## 1. Curiosity

What other forms can a programming language take?

What other notions of programming are possible?

## 2. Productivity

Procedural languages, including C, C++ and Java, are very detailed.

Many source lines imply significant development and maintenance expenses.

### 3. Reliability

Too much low-level detail in programs greatly enhances the chance of minor errors. Even minor errors can raise significant problems in applications.

### 4. Security

Computers are entrusted with ever-increasing responsibilities. How can we know that a program is safe and reliable enough to trust?

### 5. Execution speed

Procedural languages are closely tied to the standard sequential model of instruction execution. We may need radically different programming models to fully exploit parallel and distributed computers.

# Desirable Qualities in a Programming Language

Theoretically, all programming languages are equivalent (Why?)

If that is so, what properties are desirable in a programming language?

- It should be easy to use.

Programs should be easy to read and understand.

Programs should be simple to write, without unexpected pitfalls.

It should be *orthogonal*, providing only one way to do each step or computation.

Its notation should be natural for the application being programmed.

- The language should support abstraction.

You can't anticipate all needed data structures and operations, so adding new definitions easily and efficiently should be allowed.

- The language should support testing, debugging and verification.
- The language should have a good development environment.

Integrated editors, compilers, debuggers, and version control are a big plus.

- The language should be portable, spanning many platforms and operating systems.

- The language should be inexpensive to use:

Execution should be fast.

Memory needs should be modest.

Translation should be fast and modular.

Program creation and testing should be easy and cheap.

Maintenance should not be unduly cumbersome.

Components should be reusable.

# Programming Paradigms

Programming languages naturally fall into a number of fundamental styles or *paradigms*.

## Procedural Languages

Most of the widely-known and widely-used programming languages (C, Fortran, Pascal, Ada, etc.) are procedural in nature.

Programs execute statement by statement, reading and modifying a shared memory.

This programming style closely models conventional sequential processors linked to a random access memory (RAM).

Question:

Given

```
a = a + 1;  
if (a > 10)  
    b = 10;  
else b = 15;  
a = a * b;
```

Why can't 5 processors each execute one line to make the program run 5 times faster?