

Pure Functional languages (1970s–present)

Key Idea: *Referential Transparency*

- programs are expressions

Programmers give up some control—

- storage management (garbage collection)
- sequencing of operations (demand-driven evaluation)

—and shed the corresponding responsibilities.

Basic operation:

Replacement of subexpressions by their values

The underlying machine is practically invisible. Code size drops 80–90%; correctness proofs are easier.

Examples: Haskell, Miranda, Concurrent Clean

All share other amenities:

- higher-order functions
- lazy evaluation, infinite data structures
- polymorphic types
- curried functions
- clean, simple syntax

What does the future hold?

The sooner we can forget that FORTRAN ever existed, the better, for as a vehicle of thought it is no longer adequate: It wastes our brainpower, and it is too risky and therefore expensive to use. FORTRAN's tragic fate has been its wide acceptance, mentally chaining thousands and thousands of programmers to our past mistakes.

— Edsger W. Dijkstra
“The Humble Programmer”
Turing Award Lecture, 1972

Thirty years later, the same could be said about C, and every one of today's programming languages will eventually meet the same fate.

The pattern repeats endlessly:

- C and Pascal replace Fortran
- Modula-2 replaces Pascal
- C++ replaces C and Modula-2
- Java replaces C++ (?)
- ??? replaces Java

One generation's hot-air balloon becomes the next generation's anchor.

Each replacement leaves thousands of programmers chained to mistakes of the past.

Lessons for the long term:

- Don't form permanent attachments to transient entities (such as particular programming languages).
- Save your lasting commitments for things that last (such as logic and mathematics).
- Be a skeptic: A field with serious problems, and money to spend, attracts charlatans and peddlers of snake oil.
- Share your expertise and your skepticism: Help your fellow citizens understand
 - * what computing cannot do
 - * what computing should be expected to do
 - * what computing should be prevented from doing.