

Chapter 1

Introducing C

Origins of C

- C is a by-product of UNIX (**Linux by Linus Torvalds**), developed at Bell Laboratories by Ken Thompson, Dennis Ritchie, and others.
- Thompson designed a small language named B.
- B was based on BCPL, a systems programming language developed in the mid-1960s.

Origins of C

- By 1971, Ritchie began to develop an extended version of B.
- He called his language NB (“New B”) at first.
- As the language began to diverge more from B, he changed its name to C.
- The language was stable enough by 1973 that UNIX could be rewritten in C.

Standardization of C

- *K&R C*
 - Described in Kernighan and Ritchie, *The C Programming Language* (1978)
 - De facto standard
- *C89/C90*
 - ANSI standard X3.159-1989 (completed in 1988; formally approved in December 1989)
 - International standard ISO/IEC 9899:1990
- *C99*
 - International standard ISO/IEC 9899:1999
 - Incorporates changes from Amendment 1 (1995)

C-Based Languages

- **C++** includes all the features of C, but adds classes and other features to support object-oriented programming.
- **Java** is based on C++ and therefore inherits many C features.
- **C#** is a more recent language derived from C++ and Java.
- **Perl** has adopted many of the features of C.

Properties of C

- Low-level: machine-level concept including address, bitwise operators, operation close to CPU's built-in instructions (like ++, --)
- Small: to keep C small, C heavily relies on a library of standard functions. (not procedure, subroutine, method)
- Permissive: C assume you know what you are doing. You have wider degree of latitude than other languages. No mandated error-checking feature.

Strengths of C

- Efficiency
- Portability
- Power
- Flexibility
- Standard library
- Integration with UNIX

Weaknesses of C

- Programs can be error-prone.
- Programs can be difficult to understand.
- Programs can be difficult to modify.

Effective Use of C

- Learn how to avoid pitfalls.
- Use software tools (`lint`, debuggers) to make programs more reliable.
- Take advantage of existing code libraries.
- Adopt a sensible set of coding conventions.
- Avoid “tricks” and overly complex code.
- Stick to the standard.

Obfuscated C: N-Queen

```
v, i, j, k, l, s, a[99];
main()
{
for(scanf("%d",&s);*a-s;v=a[j*=v]-a[i],k=i<s,j+=(v=j<s&&
(!k&&!!printf(2+"\n\n%c"-(!l<<!j)," #Q"[l^v?(l^j)&1:2]))&&
++l||a[i]<s&&v&&v-i+j&&v+i-j))&&!(l%=s),v||(i==j?a[i+=k]=0
:++a[i])>=s*k&&++a[--i])
;
}
```