

1961



The unknown pop group, The Beatles, performed at the Cavern Club in Liverpool.

1961



The Russian cosmonaut, Yuri Gagarin, was the first man in space as he orbited the earth in the spaceship Vostok.

1962



The New York pop artist, Andy Warhol, increased Marilyn Monroe's fame with his series of portraits produced shortly after her death.

1964



The IBM system/360 developed by the International Business Machines Corporation in New York became the basis for their own computers as well as those developed by other companies.

1967



More than half of all Americans were against continuing the Vietnam War and took part in anti-war demonstrations.

_first breakthroughs

Equipped with the experiences of ALGOL 60, Dijkstra applied himself to **multiprogramming**. He developed a set of methods and algorithms now considered fundamental in **operating systems**:

- **semaphores** and the **mutual exclusion problem** for n processes
- the problem of "deadly embrace" and the so called **Banker's algorithm** to solve the problem

The essential parts of his work in multiprogramming were summarized in "**Cooperating Sequential Processes**" ('68), generally regarded as the starting point for the field of concurrent programming. Already available in 1965 as a manuscript, it served as lecture notes for a course and included the inter-process communication problem, the **Sleeping Barber Problem**; and the problem of the **Dining Quintuple** – later dubbed the **Dining Philosophers** by C.A.R. Hoare – was the examination question at the end of this course.

His experience in process synchronization and deadlocks was the base for the first **layered operating system**, named "THE". The operating system was organized as a hierarchy of layers, each one constructed upon the one below it. This work introduced design principles which have become part of the working vocabulary of every professional programmer, like **level of abstraction** or **layered architecture**.

A further Dijkstra cornerstone: the **abolishment of the GOTO statement** formulated in 1968 in a letter to the ACM, whose editor then produced the famous title "Go-to statement considered harmful". In 1969 Dijkstra wrote his "**Notes on Structured Programming**" which became a sensation, and major corporations initiated programs based on his ideas to integrate rigorous practices into their programming projects.

Leaving NDRE for NCC Nygaard had dealt with **simulations** of different tasks, which resulted in 1961 in the start of the Simula I development, which Nygaard, and later on also **Dahl**, accompanied through four stages:

- **1961-1962**: a mathematically formulated "discrete event network" concept and thoughts on a description language
- **1962-1963**: increased flexibility due to the possibilities of ALGOL 60, with a Simula compiler as a **pre-processor to ALGOL 60**
- **1963-1964**: implementation of Simula I through an extension of UNIVAC's ALGOL 60 compiler, based upon a new **storage management scheme** developed by Dahl. The pre-processor idea was dropped
- **1964**: implementation of the **Simula I compiler**

ALGOL 60 was chosen for its block structure, good programming safety and for European patriotism. Note the important transition between stage two and three above. ALGOL's strict dynamic stack regime of procedure calls and storage allocation was inadequate for sophisticated simulation facilities, where objects tend to behave according to queuing principles. In 1965 and early '66 the discovery of language shortcomings prompted thoughts about a revision. An example:

"...processes often shared a number of common properties, both in data attributes and actions, but were structurally different in other respects so that they had to be described by separate declarations. Such partial similarity fairly often applied to processes in different simulation models, indicating that programming effort could be saved by somehow pre-programming the common properties."

The work to resolve this problem resulted in the **class/subclass concepts** which structured the rest of the new language. "...we decided that:

- We would design a new general programming language, in terms of which an improved Simula I could be expressed.
- The basic concept should be classes of objects.
- The prefix feature, and thus the subclass concept, should be a part of the language.
- Direct, qualified references should be introduced."

This was the starting point of Simula 67, the first object-oriented programming language. In addition to the technological work Nygaard began to reason about the **social implications of information technology**. In the beginning of 1967 Nygaard contacted the Trade Unions and proposed building up competences in information technology.

