DAHL66
Dahl,O. and Nygaard,K.;
SIMULA-An Alogol-Based Simulation Language;
Communications of the ACM, 9,9 (September 1966) pp. 671-678.
This paper presents the principal features of the language SIMULA. SIMULA is a true extension of ALGOL-60. The language was designed to serve the dual purpose of system description and simulation programming of discrete event system. In factt, the system description serves as a source language simulation program.

The authors make the claim that it is natural to let a simulation language contain an algorithmic language as a subset as there is, already, the need for having algorithmic procedures as part of a discrete event system description. Hence, their prime design objective was to make SIMULA not only a logical extension of an algorithmic language. Thus, SIMULA contains ALGOL-60 as a subset.

Like ALGOL, Simula has blocks, each being a set of procedures and data declarations and a sequence of operations on the data. However, SIMULA extends on ALGOL. It includes the notion of a collection of such blocks, called processes, conceptually operating in parallel. The processes perform their operations in a group called active phases or events. Interleaved between any two consecutive events of one process, any number of events of other processes may occur. Processes may be activated, suspended, passive, and terminated. Thus, SIMULA supports the concept of coroutines. The sequence of operations in the system is the sequence of active processes present in the system.

An activity is a class of process described by the sam declaration. A process is a dynamic instance of an activity declaration. It is generated by evaluating an expression with an activity identifier; a procedure name. Processes are referenced indirectly by items called elements. A mechanism, called remote accessing, is provided to make the attributes of a process accessible from the outside. An ordered sequence of elements is called a set. The element and set concepts serve to facilitate and standardize the manipulation of queues and other linear list of processes. They are also the means of delimiting and sequencing elements of processes, with or without reference to the concept of system time.

To cap off the presentstion the authors elaborates on two example SIMULA descriptions. The first example is a job shop system. The second example is a simple epidemic model.

This paper describes a fairly early version of SIMULA - SIMULA I. The class concept, for which SIMULA is most famous for, is not part of this particular version of SIMULA. However, we can identify traces of the class concept in SIMULA I. The generalization of the notion of block to a class did not appear until SIMULA 67. Also, the subclass concept is clearly not present in SIMULA I.