

INTERNATIONAL STANDARD

ISO/IEC
8631

Second edition
1989-08-01

Information technology — Program constructs and conventions for their representation

*Technologies de l'information — Structures de programmes et normes pour leur
représentation*



Reference number
ISO/IEC 8631 : 1989 (E)

Contents

	Page
Foreword	iii
Introduction	iv
1 Scope	1
2 Definition of program construct	1
3 How constructs may be combined	1
4 Specification of constructs	1
4.1 Imperative construct	1
4.2 Serial construct	1
4.3 Parallel construct	1
4.4 Iterative construct	1
4.5 Selective choice construct	1
5 Termination	2
6 Definition of subsets	2
Annex	
A Charting notations for program constructs	3

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) together form a system for worldwide standardization as a whole. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for approval before their acceptance as International Standards. They are approved in accordance with procedures requiring at least 75 % approval by the national bodies voting.

International Standard ISO/IEC 8631 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

This second edition cancels and replaces the first edition (ISO 8631 : 1986), of which it constitutes a minor revision.

Annex A of this International Standard is for information only.

Introduction

It is accepted that a limited number of distinct constructs combined in a well-defined manner is sufficient to express any process. A program is considered to be well-structured if it is built from the constructs contained in this International Standard and follows the rules of combination.

A program may be viewed at several conceptual levels. At any but the lowest level, one construct may be represented as a number of constructs at a lower level.

Information technology — Program constructs and conventions for their representation

1 Scope

This International Standard is concerned with the expression of procedure oriented algorithms. It

- a) defines the nature of program constructs;
- b) indicates the manner in which constructs can be combined;
- c) provides specifications for a set of constructs;
- d) permits the definition of a variety of subsets of the defined constructs.

See annex A for symbolic representations.

2 Definition of program construct

A program construct consists of a set of one or more procedure parts and a control part which may be implicit.

Each procedure part consists of one or more operations to be performed or may be null.

The control part determines the manner in which the procedure parts are to be executed. It can consist of a directive and a set of conditions. The control part then activates or de-activates the procedure part(s) depending on the nature of the directive and the values of the conditions. If there is neither directive nor condition, control is called implicit.

3 How constructs may be combined

The only way in which constructs can be combined to build a well-structured program is by replacing a procedure part of one construct by a complete construct.

4 Specification of constructs

4.1 Imperative construct

This construct contains one procedure part and an implicit control part which determines that the procedure part is executed exactly once.

4.2 Serial construct

This construct contains two or more procedure parts and an implicit control part which determines that the procedure parts are to be executed exactly once in the sequence given.

4.3 Parallel construct

This construct consists of two or more procedure parts and a control part which initiates these procedure parts. Execution of the construct is finished when all initiated procedure parts are completely executed.

4.4 Iterative construct

- a) Pre-tested iteration

This construct consists of a procedure part and a control part with one condition, the value of which determines whether the procedure part is executed zero or more times.

- b) Post-tested iteration

This construct consists of a procedure part and a control part with one condition, the value of which determines whether the procedure part is executed more than once.

- c) Continuous iteration

This construct consists of a procedure part and a control part with an implicit condition which specifies that the procedure part will be repeated indefinitely.

4.5 Selective choice construct

- a) Monadic selective

This construct consists of a single procedure part and a control part with one condition, the value of which determines whether or not the procedure part is to be executed.

- b) Dyadic selective

This construct consists of two procedure parts and a control part with one condition, the value of which determines which one of the two procedure parts is to be executed.

c) Multiple exclusive selective

This construct consists of a number of procedure parts and a control part with a set of conditions, the values of which determine which one of the procedure parts is to be executed.

d) Multiple inclusive selective

This construct consists of a number of procedure parts and a control part with a set of conditions, the value(s) of which select zero or more procedure parts to be executed in an undefined sequence.

5 Termination

In addition to the termination of a construct as defined by its control part, the execution of a construct may be terminated by a TERMINATION operation placed in one or more procedure parts of the construct. The TERMINATION operation shall identify which construct is to be terminated. If the

TERMINATION operation is executed, execution of the identified construct and all its inner constructs will immediately cease.

A TERMINATION operation that would terminate a parallel construct or a multiple inclusive selective construct is undefined.

A TERMINATION operation which terminates an outer construct does not conform to this International Standard.

6 Definition of subsets

The use of a proper subset of the constructs defined in this International Standard and combined in accordance with this International Standard shall be considered to be in conformance with this International Standard.

Use of a construct other than the ones defined in this International Standard which is functionally equivalent to a legitimate composition of constructs defined in this International Standard is in conformance with this International Standard.

Annex A

(informative)

Charting notations for program constructs

The following charting notations for program constructs in columns A to G are examples of applicable graphic representations.

The "Reference" column uses the symbols of ISO 5807 and is included for reference only.

Symbols from the various columns should not be intermixed.

CHARTING NOTATIONS FOR PROGRAM CONSTRUCTS

Construct	REFERENCE	A	B	PSD	PROGRAM STRUCTURE DIAGRAMS	DSD	DESIGN STRUCTURE DIAGRAMS	C	SPD	STRUCTURED PROGRAMMING DIAGRAMS	D	HCP	HIERARCHICAL AND COMPACT DESCRIPTION CHART	E	PAD	PROBLEM ANALYSIS DIAGRAMS	F	LCP	LOGICAL CONCEPTION OF PROGRAM HIERARCHICAL	G	LCP	LOGICAL CONCEPTION OF PROGRAM FLOW CHART	H	R	CHARTS
5.1	Imperative																								
5.2	Serial																								
5.3	Parallel																								


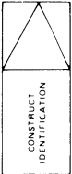


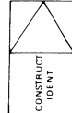


CHARTING NOTATIONS FOR PROGRAM CONSTRUCTS
ITERATIVE CHOICE CONSTRUCTS

REFERENCE	A	B	C	D	E	F	G	H	
PF PROGRAM FLOWCHARTS	PSD PROGRAM STRUCTURE DIAGRAMS	DSO DESIGN STRUCTURE DIAGRAMS	SPD STRUCTURED PROGRAMMING DIAGRAMS	HCP HIERARCHICAL AND COMPACT DESCRIPTION CHART	PAD PROBLEM ANALYSIS DIAGRAMS	LCP LOGICAL CONCEPTION OF PROGRAM HIERARCHICAL	LCP LOGICAL CONCEPTION OF PROGRAM FLOW CHART	R CHARTS	
Construct 5.4 a Pre Tested Iteration									
5.4 b Post Tested Iteration									
5.4 c Continuous Iteration									

CHARTING NOTATIONS FOR PROGRAM CONSTRUCTS
SELECTIVE CHOICE CONSTRUCTS

Construct	REFERENCE	PF PROGRAM FLOWCHARTS	A	PSD PROGRAM STRUCTURE DIAGRAMS	B	DSD DESIGN STRUCTURE DIAGRAMS	C	SPD STRUCTURED PROGRAMMING DIAGRAMS	D	HCP HIERARCHICAL AND COMPACT DESCRIPTION CHART	E	PAD PROBLEM ANALYSIS DIAGRAMS	F	LCP LOGICAL CONCEPTION OF PROGRAM METALOGICAL	G	LCP LOGICAL CONCEPTION OF PROGRAM FLOWCHART	H	R CHARTS
5.5 a	Multiple Selective																	
5.5 b	Divide Selective																	
5.5 c	Multiple Exclusive Selective																	
5.5 d	Multiple Inclusive Selective																	

CHARTING NOTATIONS FOR PROGRAM CONSTRUCTS

Construct	REFERENCE	A	B	C	D	E	G	H
	PF PROGRAM FLOWCHARTS	PSD PROGRAM STRUCTURE DIAGRAMS	DSD DESIGN STRUCTURE DIAGRAMS	SPD STRUCTURED PROGRAMMING DIAGRAMS	HCP HIERARCHICAL AND COMPACT DESCRIPTION CHART	PAD PROBLEM ANALYSIS DIAGRAMS	LCP LOGICAL CONCEPTION OF PROGRAM	R CHARTS
6								
Member bodies proposing and maintaining the sets of symbols	ISO WD	NETHERLANDS NATIONAL STANDARD NEN 1422 GERMANY NATIONAL STANDARD DIN 66261	UNITED KINGDOM BRITISH STANDARD BS 6724	JAPAN JIS X 0040:1986 (JIS X 0040:1986) JIS X 0040:1986 (JIS X 0040:1986)	INDONESIA JAWA PURBA JIS X 0040:1986 (JIS X 0040:1986)	AFNOR 267 102		GOST 19.005-85
Notes	THIS SET OF SYMBOLS IS FOR REFERENCE AND COMPARISON ONLY	1. A COMPLETE PROGRAM, A CONSTRUCT OR A PROCEDURE PART ALL HAVE THE OUTLINE OF A RECTANGLE. THIS SECTION 4 OF THE STANDARD IS ACHIEVED ENTIRELY BY THE USE OF RECTANGLES. NO CONNECTING FLOW LINES ARE USED. 2. IN ALL SELECTIVE CHOICE SYMBOLS, THE LOWER POINT OF THE RECTANGLE IS THE POINT OF ENTRY. THE POINT WITH EITHER OF THE TWO SIDES OF THE CONSTRUCT OPERATION MUST REPLACE A COMPLETE PROCEDURE PART OF A CONSTRUCT.					1. ALL THE REPRESENTATIONS ARE ALSO ABLE TO DESCRIBE THE DATA TO BE OBTAINED OR TO BE USED. 2. THE SYMBOL (C) OF STRUCTURE 5.3 INDICATES THE INDEPENDENCE OF THE PROCEDURE BEING CARRIED OUT SIMULTANEOUSLY THE SYMBOL (A) OF STRUCTURE 5.6 INDICATES THE EXCLUSION OF THE PROCEDURE. 3. THE REPRESENTATION OF AN ENTIRE PROGRAM OR DATA SET IS CARRIED OUT BY INTERMIXING THESE STRUCTURES.	

UDC 681.3.06

Descriptors : data processing, programming (computers), computer programs.

Price based on 7 pages
