

Lib sc. 9;1972; PAPER B.

Computer Programming Language: Depth Classification Version of CC.
(Classification problems. 55). (Design series. 27).

I K Ravichandra Rao and Kanti Rawat*, *Documentation Research and Training Centre, Indian Statistical Institute, Bangalore 3.*

[A depth classification version of Colon Classification for compound subjects going with the Host Subject "9S,8 Computer Programming Language" is given. The methodology of design is based on that for freely faceted classification. Computer Programming Language is deemed to be a Personality Isolate going with the Main Subject "9S Computer Science". The principles used in arranging the speciators derived on the basis of different quasi-isolates are mentioned. An Index to the schedule, a list of twenty-four examples classified according to the depth version, and an alphabetical index to the subjects in the classified list of examples, are given.]

ABBREVIATIONS USED

(BS) = Basic Subject
(E) = Energy Isolate
(M) = Matter Isolate

(P) = Personality Isolate
(QI) = Quasi Isolate

1 Scope of the Paper

This paper demonstrates the design of a depth classification version of CC for compound subjects going with the Host Subject "9S,8 Computer Programming Language". The methodology for designing a freely faceted scheme for classification based on postulates, canons, and principles, has been used (4, 5).

2 Definition

21 REFERENCE BOOKS

The following publications were helpful as sources of definition of terms and for selection of isolates relating to "program"

* Present Address: Computer Science Unit, Indian Statistical Institute, Calcutta 35.

and "programming language":

- 1 SAMMET (J E). Programming languages: History and fundamentals. 1969.
- 2 BS 3527:1962. BRITISH STANDARDS (— Institution). Glossary of terms used in automatic data processing. 1962.
- 3 AMERICAN STANDARDS (— Association). American standard vocabulary for information processing. 1966.
- 4 HAYES (R M) and BECKER (J). Handbook of data processing for libraries. 1970. P 790-813.

For convenience, two of the terms are defined in the succeeding sections.

22 COMPUTER PROGRAM

A "Computer program" is a sequence of instructions by which a computer will execute a desired task.

23 COMPUTER PROGRAMMING LANGUAGE

A "Computer programming language" is a language used for the preparation of computer program.

3 Basic Subject and Host Subject

31 PROGRAMMING LANGUAGE

The idea "design and development of programming languages" is deemed to be an isolate going with the (BS) Computer Science.

32 "COMPUTER SCIENCE" AS MAIN SUBJECT

321 Scope

It is found helpful to deem "Computer Science" a Main Subject. The developments in the subject in recent years and the literary warrant support this. The following is a statement from a recent commentary by Pylyshyn (3):

"Computer science is an emerging discipline. Its boundaries are not clearly defined, for it overlaps many established disciplines. Computer scientists . . . conduct research and publish papers in engineering, mathematics, economics, sociology, psychology, linguistics, philosophy, library science, the biological sciences, business, law, and the humanities, as well as in other cross-discipline areas such as communications and information theory, control and general systems theory.

"We are not referring here to the disciplines in which the computer has found fruitful applications as a tool. We are referring to those areas whose subject matter has become a legitimate concern of people who call themselves *computer scientists*. For example, (to take a very

arbitrary sample of computer science problems), W S McCulloch has been concerned with the properties of neurons and nerve nets, A Turing with certain problems in formal logic, Noam Chomsky with formal linguistic systems, Herbert Simon with the nature of the problems-solving process in humans, John von Neumann with comparisons between the computer and the brain, and Norbert Wiener and W R Ashby with the relevance of computer-science principles to neurology and psychiatry as well as with comparisons of control processes in biological and mechanical systems"

322 *Origin*

The counter, the adder and the calculator preceded the computer many centuries earlier. They were themselves preceded by the tradition of reckoning. This tradition, in turn, is a result of the convergence of two separate traditions, the tradition of counting and the tradition of using symbol designations for real things (6).

323 *Position of Main Subject*

On the basis of the above description of computer science, which has elements taken from several subjects in the natural sciences, and in the humanities and social sciences, the fact of the use of computer in many subject-fields, and considering the historical origins of computer, it is helpful to place the new Main Subject earlier to "A Natural Sciences". Provisionally the Main Subject Number "9S" has been assigned to "Computer Science". Thus, we have

- 9P Communication theory
- 9Q Symbolism
- 9S Computer science
- A Natural Sciences (Partial Comprehension)
- AZ Mathematical Sciences (Partial Comprehension)
- B Mathematics

33 **HOST SUBJECT**

It is found help to deem "programming language" as an isolate in (IPI) going with the Main Subject "9S Computer Science". A few of the provisional isolates in (IPI) are as follows:

- 2 Automata theory
- 3 Program
- 5 Programming language

34 SEQUENCE OF ISOLATES

Some of the principles used in arranging the isolates in the schedules for (IP1) and those for (IM1) are mentioned in the succeeding sections.

341 *Principle of Later in Time*

The Principle of Later in Time has been more or less conformed to in arranging the ideas derived on the basis of the following (Q1) in (IP1):

- CV3 By Display equipment
- G By Grammar
- L By Type of language
- P By Class of language

342 *Principle of Increasing Complexity*

The Principle of Increasing Complexity has been used in arranging the ideas derived on the basis of the (Q1) By Form of display in (IP1).

343 *Principle of Scheduled Mnemonics*

The sequence of ideas derived on the basis of the (Q1) By Function in (IP1) is parallel to a similar schedule prepared earlier for subjects going with the Host Subject "Computer Production" (1).

344 *Specialist Preference*

In (IM1) the sequence of the (Q1) and that of the ideas derived on the basis of each (Q1) generally conform to the sequence of such ideas usually preferred by specialists in programming languages.

4 Schedule of Isolates

41 SPECIATORS IN (IP1)

The Quasi Isolates helpful in deriving speciators for forming Compound Isolates in (IP1) — that is, variety of computer languages — are enumerated in Table 1 in Sec 411. The speciators have been selected by blending the *a priori* and pragmatic approaches. The latter consisted in a study and abstracting of about three hundred selected recent microdocuments — such as, articles in periodicals, and abstracts of articles — and perusal of selected treatises and glossaries pertaining to the subject. The sequence among the (Q1), determined by applying Group Strategy, Wall-Picture Principle, and other Principles for Helpful Sequence, is deemed to be helpful to a majority of the specialists in the subject (2).

411 Table 1. Quasi Isolates in (1P1)

SN	Sector (S—)	Quasi Isolate
(a)	(b)	(c)
1	(...)	By Name
2-3	R1Z	By Conversion and compatibility
2	R3	By Type of compatibility
3	R2	By Ease of conversion
4	R0	By Kind of computer in which usable
5	P	By Class of language
6	L	By Type of language
7	G	By Grammar
8	C	By Purpose
9	9A	By Functional characteristics
10	OZ	By Stage

42 SCHEDULE OF (1M1)

A schedule of special Property Isolates is given in (1M1). Some of the isolates are subdivided by using appropriate characteristics. A compound Matter-Property isolate may be formed by compounding two or more isolates taken from this schedule, the components being connected by "hyphen" (-). Isolates from the schedule of Common Property Isolates can be used wherever found necessary.

43 SCHEDULE OF (1E)

Isolates from the schedule of Common Energy Isolates can be used wherever found necessary.

5 Allocation of Notation

The sectors allocated to the (QI) in (1P1) are indicated in col b of Table 1 in Sec 411.

6 Index to Schedule

Note.— 1 The terms enumerated in the schedules in Sec 7 are listed in this index. However, terms denoting ideas the numbers for which are indicated to be derived by using such devices as (AD), (ND), (SD) and "Division by" are not included.

2 The number from the schedule given against each index entry is preceded by an abbreviation for the name of the appropriate

Fundamental Category — for example, (IP1), (IM1), and (IE).

- Accumulator (IM1), 8AJE
- Address
 - Calculator (IP1), CX4
 - Translator (IP1), C823
- Addressing (IM1), 8AJA
- Algebraic (IM1), 745
 - expression manipulator statement (IM1), 871
- Algorithmic language (IP1), LM
- Alphanumeric
 - character (IM1), 1252
 - data (IP1), CDIG2
 - description (IM1), 4352
 - handling (IM1), 82
 - designational expression (IM1), 42452
 - identifier (IM1), 16222
 - program unit label (IM1), 16452
- Alphabetic
 - character (IM1), 1251
 - data (IP1), CDIG1
 - description (IM1), 74351
 - designational expression (IM1), 42451
 - identifier (IM1), 16221
 - program unit label (IM1), 16451
 - variable (IM1), 411
- And (IM1), 4B41
 - link deletion (IM1), 14222
- Application (QI), (IP1), CSZ
- Arithmetic (IM1), 741
 - expression (QI), (IM1), 481
- Array (IM1), 41F
 - declaration (IM1), 72
- Arrests (IM1), 1281
- Assembler (IP1), 183
- Assembling (IM1), 073
 - language (IP1), P4
- Associativity (IM1), 633
- Backus
 - Naur form (IP1), G5
 - normal form (IP1), G5
- Basic element (IM1), 14
- Binary (IM1), 62C
 - coded decimal (IM1), 62C1
- Biquinary (IM1), 62C2
- Blank column (IM1), 7815
- Block statement (IM1), 81
- Boolean
 - expression (qi) (IM1), 484
 - iri Data description (IM1), 742
 - function (IM1), 4235
 - query (IP1), CT482
 - type declaration (IM1), 71C
 - value (IM1), 413
- Bounded context
 - grammar (IP1), GN
 - translation (IP1), P62
- Branch statement (IM1), 83121
- Built-in-function (IM1), 4231
- Categorical grammar (IP1), CQ
- Cathode ray tube *irt*
 - Equipment used (IP1), CV38
- Output statement (IM1), 8A28
- Cell
 - declaration (IM1), 71E
 - designation (IM1), 426
- Character *irt*
 - Data description (IM1), 743
 - Form of the display (IP1), CV6
 - Functional form of language (IM1), 12
- Class (QI), (IP1), P
 - Closed (IM1), 4211
- Cluster research (IP1), CT45
- Colon (IM1), 1285
- Comma (IM1), 1287
- Command (IM1), 65
- Combination of variable and constant (IM1), 74B
- Commercial application (IM1), C9X
- COMMON (IM1), 7A1
- Comparison (IM1), 14226
- Compatibility and conversion (QI), (IP1), R1
- Compiler (IP1), C82
 - directive (IM1), 68
 - independence (IP1), R34
- Compiling (IM1), 076
 - language (IP1), P6
- Complete grammar (IP1), G4
- Complex (IM1), 744
 - assignment statement (IM1), 8E
 - constituent (IPD), GP6
- Compound (IP1), CD13
 - data structure (IM1), 622
 - statement (IM1), 81
- Computed go to statement (IM1), 8312
- Computer compatibility (IP1), R21
- Communication (IM1), 075
- Commutativity (IM1), C32
- Conditional (IM1), 8311
- Constant and variable (IM1), 74B
- Consistency (IM1), 03
- Console typewriter *irt*
 - Equipment used (IP1), 3V341
 - Machine feature (IM1), 8AJF1

- Consolidator (IP1), C84
- Context
 - free grammar (IP1), GD
 - phrase structure grammar (IP1), GP4
 - sensitive (IP1), GP2
- Control (IP1), CF
 - console *irt*
 - Input statement (IM1), 8A11
 - Output statement (IM1), 8A51
 - statement (IM1), 831
- Conversion
 - and compatibility (QD), (IP1), R1
 - irt* Alphanumerical data handling (IM1), 826
 - Efficiency (IM1), 078
- Cycling system (IM1), 832
- Data
 - analysis (IP1), CD8
 - and its description (IM1), 62
 - collection (IP1), CD2
 - communication (IP1), CDA
 - computation (IP1), CDC
 - correlation (IP1), CD6
 - description (IM1), 74
 - handling (IM1), 074
 - manipulation (IP1), CD
 - name (IM1), 1622
 - processing (IP1), CDA
 - reduction (IP1), CD4
 - store statement (IM1), 7A0
- Debugging statement (IM1), 8AD
- Decision making and sequence control (IM1), 83
- Declaration (IM1), 66
 - and non-executable statement (QD) (IM1), 7
- Deodecimal (IM1), 62M
- Design (IP1), CH
- Designational expression (QD) (IM1), 42
- Dialect (IP1), R35
- Dictionary display (IP1), CT48
- Digit (IM1), 412
- Dilimeter (IM1), 6B
- DIMENSION (IM1), 7A2
- Direct addressing (IM1), 8AJA1
- Discontin constituent (Ipd), GPS
- Display (IP1), CV
- Do statement (IM1), 83121
- Double
 - inverted comma (IM1), 1282
 - precision (IM1), 4B17
- Dual fixed point *irt*
 - Arithmetic (IM1), 7414
 - expression (IM1), 4B14
- Dummy statement (IM1), 84
- E mode (IM1), 7814
- E notation (IM1), 7814
- Ease of conversion (QD) (IP1), R2
- Editing *irt*
 - Alphanumerical data handling (IM1), 824
 - Information processing (IP1), CT8
- Efficiency (IM1), 07
- Electrical signal *irt*
 - Input statement (IM1), 8AJA
 - Output statement (IM1), 8A5A
- Equipment
 - and operating system (IM1), 8A
 - or operating system (IM1), 8A
 - used (QD) (IP1), CV3
- Equivalence (IM1), 7A3
- Equivalent (IM1), 4B42
- Error condition statement (IM1), 833
- Executable statement (IM1), 8
- Expression (QD) (IM1), 4
- Experimental (IP1), 1
- Extension (IP1), R38
- External (IM1), 736
- F-notation (IM1), 7812
- False (IM1), 4132
- Finite state grammar (IP1), GB
- File description (IM1), 76
- Fixed point *irt*
 - Arithmetic (IM1), 7413
 - expression (IM1), 41513
- Array declaration (IM1), 723
- Input (IM1), 7813
- Type of declaration (IM1), 712
- Variable (IM1), 41L
- Flexible (IM1), 4217
- Floating point *irt*
 - Arithmetic (IM1), 7412
 - expression (IM1), 4B12
- Array declaration (IM1), 722
- Input (IM1), 7812
- Type of declaration (IM1), 711
- Variable (IM1), 41J
- For statement (IM1), 8322
- Four address (IM1), 8AJ4
- Form (QD) (IM1)
 - of data (QD) (IP1)
 - the display (QD), (IP1)
- Formal (IM1), 745
- grammar (IP1), G4

- Format description (IM1), 78
 Formation rule (IM1), 166
 Full stop (IM1), 1284
 Fully developed (IP1), 7
 Function *irt*
 Designational expression (IM1), 423
 Purpose (QI), (IP1)
 Functional characteristic (QI) (IM1), 009 (IP1), 9A
 form of the language (IM1), 1
 part of the language (IM1), 6
 statement (IM1), 8c
 General
 literal (IM1), 181
 purpose (IP1), C2
 Generative grammar (IP1), GC
 Go to statement (IM1), 83121
 Government affair (IP1), C9W
 Grammar (QI) (IP1), G
 Graph processing (IP1), CR8
 Graphic (IP1), CV8
 equipment (IP1), CU32
 operator (IM1), 1422
 punctuation symbol (IM1), 1426
 Guidance (IP1), Cg
 Hand written letter and numeral (IM1), 8A1B
 Hardware language (IP1), LR
 Hexadecimal (IM1), 62N
 Hierarchical
 data description (IM1), 748
 form of display (IP1), CUE
 High level language (IP1), LU7
 Hollerith (IM1), 182
 literal (IM1), 781H
 record (IM1), 781H
 string (IM1), 781B
 Hyphen (IM1), 1288
 I-notation (IM1), 7813
 Identifier (IM1), 162
 Identity (IM1), 634
 IF statement (IM1), 83111
 Imaginary part *irt*
 Complex assignment statement (IM1), 8E2
 Data description (IM1), 7442
 Immediate constituent grammar (IP1), GP4
 Implies (IM1), 4B43
 Indirect address (IM1), 8AJA2
 Inflexible (IM1), 4211
 Information
 processing (IP1), CT
 retrieval (IP1), CT4
 storage (IP1), CT3
 Input (IM1), 781
 language (IP1), C811
 statement (IM1), 8A1
 output system (IM1), 8A0Z
 Integer (IM1), 412
 type *irt*
 Arithmetic (IM1), 7413
 expression (IM1), 4B13
 Array declaration (IM1), 723
 Input (IM1), 7813
 Type of declaration (IM1), 712
 mode variable (IM1), 412
 Intermediate level language (IP1), LU4
 Internal (IM1), 731
 machine language (IP1), LB1
 Intersection (IM1), 14225
 Inverse (IM1), 635
 Iterative statement (IM1), 832
 Key word (IM1), 1424
 Kind of
 computer (QI), (IP1), R0
 data (QI), (IP1)
 Kth way (IP1), CS6K
 Label (IM1), 424
 Language (IP1), c81
 L like (IP1), R36
 Left parenthesis (IM1), 128A
 Letter (IM1), 411
 Library
 irt Assembling language (IP1), P46
 Compiling language (IP1), P66
 Translating language (IP1), P66
 reference statement (IM1), 8AC
 Light pen and sketch pad (IM1), 8A18
 Linear array (IM1), 41E
 grammar (IP1), GJ
 query (IP1), CT4B1
 Linguistic characteristic (QI) (IM1), 091
 operation (IP1), CP

- List**
 handling statement (IM1), 872
 of Pointer (IM1), 747
 processing (IP1), CR2
- Listing (IP1), CT6**
- Literal *irt***
 Format description (IM1), 781L
 Functional form of language (IM1), 18
- Logical (IM1), 742**
 exclusive OR (IM1), 4B47
 expression (QI) (IM1), 4B4
 OR (IM1), 4B46
 value (IM1), 413
- Loop control statement (IM1), 832**
- Low**
 level language (IP1), LU1
 linked list structure language (IP1), LU12
- Machine**
 aided language (IP1), C7
 feature (QI) (IM1), 8AJ0Z
 independence (IP1), R32
 oriented symbolic language (IP1), LG
- Macro (IM1), 421**
 generable assembling language (IP1), P44
 processor (IP1), C834
- Magnetic card *irt***
 Input statement (IM1), 8A16
 Output statement (IM1), 8A56
- disc *irt***
 Input statement (IM1), 8A15
 Output statement (IM1), 8A56
- tape *irt***
 Input statement (IM1), 8A14
 Output statement (IM1), 8A54
- Mathematical**
 expression (QI) (IM1), 4B
 Designational (IM), 4234
 operator (IM1), 12813
- Media *irt***
 Input statement (QI) (IM1), 8A10Z
 Output statement (QI) (IM1), 8A5DZ
- Meta semantic language (IP1), LS**
- Method of sort (QI), (IP1), CX0Z**
- Merging (IP1), Cy**
- Micro processor (IP1), C832**
- Military application (IP1), c9MV**
- Mixed arithmetic (IM1), 7415**
 point (IM1), 4B15
- Monitor (IP1), C85**
- Monodic operator (IM1), 1421**
- Multi-access (IM1), 8AJ3**
- Multi-phase (IM1), 0762**
- Multiple address (IM1), 8AJ8**
 precision (IM1), 4B18
- Name (QI) (IP1), (...)**
- New**
 lines (IM1), 7817
 records (IM1), 7816
- Node (IM1), 14221**
- Noise word *irt***
 Functional form of language (IM1), 1B
 Variable (IM1), 41e
- Non-executable and declaration statement (QI) (IM1), 7**
- Non-numeric character (IM1), 125**
 data (IP1), CD1G
 description (IM1), 7435
 designational expression (IM1), 4245
 program unit label (IM1), 1645
- Non-procedural language (IP1), LT**
- Non-specialist (IP1), LN5**
- Non-transformational grammar (IP1), GS**
- Normal grammar (IP1), CW**
- Not (IM1), 4B45**
- Notational (IM1), 02**
- Number (IM1), 414**
- Numeral and handwritten letter (IM1), 8A1B**
- Numeric character (IM1), 121**
 data (IP1), CD1B
 description (IM1), 7431
 designational expression (IM1), 4241
 program unit label (IM1), 1641
- Octal (IM1), 628**
- One**
 or three address (IM1), 8AJ813
 two address (IM1), 8AJ812
 sided linear grammar (IP1), GJ2
 Open (IM1), 4217
- Operating system and equipment (IM1), 8A**
 and machine feature (IM1), 8AJ
 or equipment (IM1), 8A

- Operation *irt*
 - Boolean expression (Q1) (IM1), 4B4DZ
 - Efficiency (IM1), 07G
 - Functional part of the Language (IM1), 63
 - Logical expression (Q1) (IM1), 4B4DZ
- Operator precedence grammar (IP1), G7
- Output language (IP1), C815
- OR
 - irt* Logical exclusive operation (IM1), 4B47
 - Operation (IM1), 4B46
 - link deletion (IM1), 14223
- Oscilloscope (IP1), CV36
- Other application (IP1), C9 (...)
- Output (IM1), 785
 - input system (IM1), 8A0Z
 - statement (IM1), 8A5
- Paper tape *irt*
 - Input statement (IM1), 8A13
 - Output statement (IM1), 8A55
- Paragraph (IM1), 418
- Partially developed (IP1), 4
- Pattern
 - handling statement (IM1), 878
 - recognition (IP1), CL
- Perform statement (IM1), 8323
- Phrase
 - procedure (IP1), CT4
 - structure grammar (IP1), GP
- Pictorial (IP1), CVB
- Plex processing (IP1), CR7
- Plex processing (IP1), CR5
- Pointer access *irt*
 - Information retrieval (IP1), CT4F
 - Machine feature (IM1), 8AJH
 - Operating system (IM1), 8AJH
- Polyphase (IP1), CXA
- Presumptive instruction (IM1), 8G
- Printed (IM1), 8A1B
 - copy (IM1), 8A57
 - magnetic character (IM1), 8A17
- Probabilistic and weighted (IP1), GU
- Problem
 - defining (IP1), CK5
 - describing (IP1), CK1
 - oriented language (IP1), LM
 - solving (IP1), CK6
- Procedure (IM1), 422
 - declaration (IM1), 71N
 - oriented language (IP1), LL
 - statement (IM1), 85
- Processor (IP1), C83
- Program unit label (IM1), 164
- Programming language 9B, 8
- Property (Q1) (IM1), 01Z
- Pseudo
 - assembling language (IP1), 942
 - linear grammar (IP1), G14
 - machine language (IP1), LB6
- Publication language (IP1), LQ
- Punched card *irt*
 - Input statement (IM1), 8A21
 - Output statement (IM1), 8A25
- Punctuation mark (IM1), 128
- Purpose (Q1) (IP1), C
- Quarternary (IM1), 624
- Quinary (IM1), 625
- Rational weighted (IP1), GU
- Reading (IM1), 074
- Real
 - part *irt*
 - Complex assignment statement (IM1), 8E1
 - Data description (IM1), 7441
 - type *irt*
 - Arithmetic (IM1), 7412
 - expression (IM1), 4B12
 - Array declaration (IM1), 722
 - Input (IM1), 7812
 - Type of declaration (IM1), 711
 - variable (IM1), 411
- Recursive operator (IM1), 1420
- Reducible grammar (IP1), GF
- Reference language (IP1), LP
- Register (IM1), 8AJE
- Regular grammar (IP1), GB3
- Remote console *irt*
 - Equipment used (IP1), CU342
 - Machine feature (IM1), 8AJF2
- Representation (IP1), LR
- Reserve word (IM1), 17
- Right parenthesis (IM1), 1281
- Scientific application (IP1), C6
- Segment base declaration (IM1), 71L
- Segmentation and storage
 - allocat statement (IM1), 8AG
 - Selection sort (IP1), CX0
- Self
 - correlation (IM1), 07E
 - extension (IM1), 07D
 - modification (IM1), 07B

- Semantic (IM1), 094
 analysis able (IM1), 1F
 Semi-colon (IM1), 1286
 Sentence (IM1), 41A
 Sequence control and decision making (IM1), 83
 Sequential formula translation (IP1), P64
 grammar (IP1), GH
 Shifting to another language (IP1), R23
 Simple (IP1), CD11
 data structure (IM1), 621
 variable (IM1), 41E
 Simulation (IP1), CJ
 Single
 address (IM1), 8AJ1
 inverted comma (IM1), 1283
 phase (IM1), 0761
 precision (IM1), 4B16
 Sketch pad and light pen (IM1), 8AJ8
 Software development (IP1), C8
 Sorting (IM1), 822
irt Alphanumerical data handling (IP1), CX
 by merging (IP1), CX6
 Replacement (IP1), CX8
 search (IP1), CX2
 Speech recognition (IP1), CN
 Special (IP1), C4
 feature (IM1), 8AJM
 character (IM1), 127
 Specialist (IP1), LNZ
 Stage (QD, OZ)
 Starter (IM1), 128A
 Statement number (IM1), 424
 Statistical expression (IM1), 4237
 Storage
 allocation specific statement (IM1), 7A
 and segmentation allocation statement (IM1), 8AG
 type (IP1), CV37
 String
 handling statement (IM1), 876
irt Data definition (IM1), 746
 Variable (IM1), 416
 processing (IP1), CR3
 Structure (IM1), 092
 Structured data (IP1), CD16
 Subroutine (IM1), 421
 Subset (IP1), R37
 Switch (IM1), 41H
 declaration (IM1), 73
 Symbolic data
 handling statement (IM1), 87
 manipulation (IP1), CR
 Synonym declaration (IM1), 71J
 Syntax (IM1), 093
 analyser (IP1), C86
 analysis table (IM1), 1D
 directed
 compiling (IP1), P65
 documentation *irt*
 Assembling language (IP1), P6L
 Compiling language (IP1), P46
 Translating language (IP1), P46
 System defined (IM1), 142
 T-grammar (IP1), GR
 Technical characteristic (QI) (IM1), 090
 Technique (QI), (IP1)
 Teletypewriter (IP1), CV3A
 Telephone (IP1), CV3B
 Ternary (IM1), 623
 Testing (IP1), CDE
 Thesaurus proving (IP1), CK
 Three address (IM1), 8AJ3
 Three or one address (IM1), 8AJ812
 Three way (IP1), CS63
 Top down syntax analyser (IP1), C861
 Topological display (IP1), CVD
 Transformational grammar (IP1), GR
 Translating (IM1), 072
 language (IP1), P6
 to another computer language (IP1), R25
 Translation (IP1), C7
 Translator (IP1), C82
 Triplex (IM1), 623
 True (IM1), 4131
 Two
 address (IM1), 8AJ2
 grammar (IP1), GD2
 to one address (IM1), 8AJ812
 way (IP1), CX62
 Type of
 compatibility (IP1), R3
 declaration (IM1), 71
 language (QI) (IP1), L
 Typed matter (IM1), 8A1B
 Typeless (IM1), 710
 Uncondit statement (IM1), 8312
 Union (IM1), 14224

Unmodified instruction (IM1), 8G
 Unstructured data (IP1), CD17
 User
 defined (IM1), 146
 oriented language (IP1), LN
 User
 written (IM1), 4232
 language (IP1), LN
 Variable (QD) (IM1), 41

and constant type (IM1), 74B
 Voice *tr*
 Input statement (IM1), 8A1C
 Output statement (IM1), 8A5A
 Web grammar (IP1), GV
 Weith-Weber precedence form
 (IP1), G71
 Writing (IM1), 071

7 Schedule

95	Computer science	C9 (NW) Movie (Cinema)
9S, 8	Programming language	C9 (BTT) Operations research
	<i>Isolates in (IP1)</i>	C9 (Y) Sociology
0Z	<i>By Stage</i>	CD <i>By Function</i>
1	Experimental	CD <i>By Form of data</i>
4	Partially developed	CD11 Simple
7	Fully developed	CD13 Compound
		CD16 Structured
9A	<i>By Functional characteristic</i>	CD17 Unstructured
	<i>Note.—Division as</i>	<i>By Kind of data</i>
	<i>for (IM1)</i>	CD1B Numeric
	<i>(Illustrative)</i>	CD1G Non-numeric
9B125	Non-numeric character set	CD1G1 Alphabetic
9B127	Special characters	CD1G2 Alphanumeric
		CD2 Data collection
C	<i>By Purpose</i>	CD4 Data reduction
C2	General	CD6 Data correlation
C4	Special	CD8 Data analysis
		CDA Data processing
C5Z	<i>By Application</i>	CDB Data communication
C6	Scientific/technical	CDC Data computation
C7	Translation (Machine- aided language)	CDE Testing
C8	Software development	CF Control
C81	Language	CG Guidance
C811	Input language	CH Design
C815	Output language	CJ Simulation
C82	Compiler/Translator	CK Theorem proving
C823	Address Translator	CK1 Problem describing
C83	Processor/Assembler	CK5 Problem defining
C832	Micro processor	CK8 Problem solving
C834	Macro processor	CL Pattern recognition
C84	Consolidator	CN Speech recognition
C85	Monitor	CP Linguistic operation
C86	Syntax analyser	CR Symbol manipulation
C861	Top down syntax analyser	CR2 List processing
C9MV	Military application	CR3 String processing
C9W	Government affairs	CR5 Plex processing
C9X	Commercial application	CR7 Picture processing
C9 (...)	Other application	CR8 Graph processing
	<i>Note.—Division by (SD)</i>	CT Information processing
	<i>(Illustrative)</i>	CT1 Information storage
		CT4 Information retrieval

	<i>By Technique</i>	GF	Reducible grammar
CT41	Phrase Procedure/hierarchical expansion	GH	Sequential grammar
CT45	Cluster research	GJ	Linear grammar
CT48	Dictionary display	G32	One-sided linear grammar
CT481	Linear query	G34	Pseudolinear grammar
CT482	Boolean query	GL	Generative grammar
CT4D	Relational form	GN	Bounded context grammar
CT4F	Pointer access	GP	Phrase structure grammar
CT6	Listing	GP2	Context sensitive
CT8	Editing	GP4	Immediate constituent (context-free-phase structure grammar)
CV	Display	GP6	With complex constituent
CV3	<i>By Equipment used</i>	GP8	With discontinuous constituent
CV32	Graphic equipment	GQ	Categorical grammar
CV341	Console typewriter	GR	Transformational grammar (T-grammar)
CV342	Remote console	GS	Non-transformational grammar
CV36	Oscilloscope	GU	Probabilistic and weighted
CV37	Storage tube	GV	Web grammar
CV38	Cathode ray tube	GW	Normal grammar
CV3A	Teletypewriter	GX	Matrix grammar
CV3B	Telephone	GY	Complete grammar
	<i>By Form of the display</i>		
CV6	Character	L	<i>By Type</i>
CV8	Graphic	LB1	Internal machine language
CVB	Pictorial	LB4	Pseudo machine language
CVD	Topological	LB6	External machine language
CVE	Hierarchical	LG	Machine oriented symbolic language
CX	Sorting	LH	Problem oriented language (Algorithmic language)
	<i>By Method</i>	LL	Procedure oriented language
CX0Z	Selection sort	LN	User written (user-oriented) language
CX1	Sorting by search	LN2	Specialist
CX2	Address calculation	LN5	Non-specialist
CX4	Sorting by merging	LP	Reference language
CX6	Two way	LQ	Publication language
CX62	Three way	LR	Hardware language (representation)
CS63	Note.—Division by (ND) (Illustrative)	LS	Meta semantic language
CS6K	Kth way	LT	Non-procedural language
CX8	Sorting by replacement	LU1	Low level language
CXA	Poly phase	LU12	Low level linked list structure language
CY	Merging	LU4	Intermediate level language
	<i>By Grammar</i>	LU7	Higher level language
G	Formal grammar	P	<i>By Class</i>
G4	Backus normal form/Backus Naur form	P4	Assembling language
G5	Operator precedence grammar	P42	Pseudo assembling language
G7	Werth-Weber precedence form	P44	Macro-generable assembling language
G71	Finite state grammar		
GB	Regular grammar		
GD	Context free grammar		
OD2	Two grammar		

P46	Syntax-directed documentation/library	01*	<i>By Functional characteristic</i>
		01	<i>By Property</i>
P6	Compiling or translating language	02	Notational
		03	Consistency
P62	Bounded context translation	07	Efficiency
P64	Sequential formula translation	071	Writing
		072	Translating
P65	Syntax directed compiling	073	Assembling
P66	Syntax directed documentation/library	074	Reading
		075	Communication
		076	Compiling
R0	<i>By kind of computer</i>	0761	Single phase
	<i>Note.—Division as in (1P1) of depth classification schedule for computer production (1)</i>	0762	Multi phase
	<i>(Illustrative)</i>	078	Conversion
		07B	Self-modification
		07D	Self-extension
		07E	Self-correction
R3(D8, B- J1)	Serial	07G	Operation
R0 (D8, B-ZK1)	Fixed	07H	Data handling
R0 (D8, B-(91B)	IBM		
		090	<i>By Technical characteristic</i>
R1Z	<i>By Conversation and compatibility</i>	091	<i>By Linguistic characteristic</i>
R2	<i>By Ease of conversion</i>	092	Structure
R21	Computer compatibility	093	Syntax
R33	Shifting to another language	094	Semantics
R25	Translating to another computer language	1	<i>By Functional form of language</i>
	<i>Note.—Division by (AD)</i>	12	Character
	<i>(Illustrative)</i>	121	Numeric
R25AL	Translating to ALGOL	125	Nonnumeric
		1251	Alphabetic
R3	<i>By Type of compatibility</i>	1252	Alpha numeric
R32	Machine independence	127	Special character
R34	Compiler independence	128	Punctuation mark
R35	Dialect	1281	Arrester/Right parenthesis
R36	Language L-like	1282	Double inverted comma
	<i>Note.—Division by (AD)</i>	1283	Single inverted comma
	<i>(Illustrative)</i>	1284	Full stop
R36AL	ALGOL-like	1285	Colon
R37	Subset	1286	Semi-colon
	<i>Note.—Division by (AD)</i>	1287	Comma
	<i>(Illustrative)</i>	1288	Hyphen
R37AL	Subset of ALGOL	128A	Starter/left parenthesis
R38	Extension	128B	Mathematical operator
	<i>Note.—Division by (AD)</i>	14	Basic element
	<i>(Illustrative)</i>	142	System defined
R38AL	Extension of ALGOL	1420	Recursive operator
(...)	<i>By Name</i>	1421	Monodic operator
	<i>Note.—Division by (AD)</i>	1422	Graphic operator
	<i>(Illustrative)</i>	14221	Node
(A)	ALGOL	14222	AND link deletion
(C)	COBOL	14223	OR link deletion
(F)	FORTRAN	14224	Union
(F=4)	FORTRAN IV	14225	Intersection
		14226	Comparison
<i>Isolates in (1M1)</i>		1424	Keyword

1426	Graphic punctuat symbol	422	Procedure
146	User defined	423	Function
162	Identifier	4231	Built-in-function
1622	Data name	4232	User written
16221	Alphabetic	4234	Mathematical
16225	Alphanumeric	4235	Boolean
164	Program unit label	4237	Statistical
1641	Numeric	424	Statement number/label
1645	Non-numeric	4241	Numeric
16451	Alphabetic	4245	Non numeric
16452	Alphanumeric	42451	Alphabetic
166	Formation rule	42452	Alpha numeric
17	Reserve word	426	Cell designation
18	Literal		
181	General	4B	<i>By Mathematical expression</i>
182	Hollerith	4B1	<i>By Arithmetic expression</i>
1B	Noise word	4B12	Floating point type real
1D	Syntax analysis table	4B13	Fixed point type/integer type
1F	Semantic analysis table	4B14	Dual fixed point
		4B15	Mixed point
4	<i>By Expression</i>	4B16	Single precision
41	<i>By Variable</i>	4B17	Double precision
-411	Alphabetic (letter)	4B18	Multiple precision
	<i>Note.—Division by sche-</i>		
	<i>dule of language isolate.</i>	4B4	<i>By Boolean expression/logical</i>
	<i>(Illustrative)</i>		<i>expression</i>
-41112	Latin		
-411142	Greek	4B40Z	<i>By Operation</i>
-411142	Russian	4B41	And
412	Digits/integer	4B42	Equivalent
-413	Logical value/Boolean value	4B43	Implies
4131	True	4B45	Not
4132	False	4B46	OR (Logical)
-414	Number	4B47	OR (Logical exclusive)
416	String		
418	Paragraph	6	<i>By Functional part of the</i>
-41A	Sentence		<i>language</i>
-41C	Noise word	62	Data and its description
-41E	Simile variable linear array	621	Simple data structure
-41F	Array	622	Compound data structure
	<i>Note.—Division by (ND)</i>		
	<i>(Illustrative)</i>		<i>By Form</i>
-41F1Z	Multi-dimensional array	62C	Binary
41F2	Two dimensional array	62C1	Binary coded decimal
-41F3	Three dimensional array	62C2	Biquinary
-41F8	Eight dimensional array	623	Ternary/Triplex
41FK	K-dimensional array	624	Quaternary
41H	Switch	625	Quinary
-41J	Floating point type variable	628	Octal
	real variable	62L	Decimal
-41L	Fixed point type variable/	62M	Duodecimal
	integer mode variable	62N	Hexadecimal
		63	Operation
42	<i>By Designational expression</i>		
421	Subroutine/macro	630Z	<i>By Axiom</i>
4211	Closed/inflexible	631	Closure
-4217	Open/flexible	632	Commutativity

633	Associativity	7814	E notation/E mode
634	Identity	7815	Blank column
635	Inverse	7816	New records
65	Command	7817	New lines
66	Declaration	7818	Hollerith string
68	Compiler directive	781H	Hollerith record/Hollerith
6B	Dilimeter	781L	literal
		785	Literal
7	<i>By Declaration and non-executable statement</i>		Out-put
71	Type of declaration		<i>Note:—Division as far "781</i>
710	Typeless	7A	Input "
711	Real/Floating point type	7A0	Storage allocation specification statement
712	Integer/Fixed point type	7A1	Data store statement
71C	Boolean type	7A2	COMMON
71E	Cell declaration	7A3	DIMENSION
71G	Function declaration	8	EQUIVALENCE
71J	Synonym declaration	81	<i>By Executable statement</i>
71L	Segment base declaration	81	Compound statement/block statement
71N	Procedure declaration	82	Alpha numerical data handling
72	Array declaration	822	Sorting
722	Floating point/real type	824	Editing
723	Fixed point/integer type	826	Conversion
73	Switch declaration	83	Sequence control and decision making
731	Internal	831	Control statement
736	External	8311	Conditional
74	Data description	83111	IF statement
741	Arithmetic	83112	Computed 'go to' statement
7412	Floating point/real type	8312	Unconditional
7413	Fixed point/integer type	83121	Go to statement
7414	Dual fixed point	83122	Branch statement
7415	Mixed arithmetic	832	Loop control statement/iterative statement/cycling system
742	Logical (= Boolean)	8321	DO statement
743	Character	8322	FOR statement
7431	Numeric	8323	PERFORM statement
7435	Non-numeric	833	Error condition statement
74351	Alphabetic	84	Dummy statement
74352	Alpha numeric	85	Procedure statement
744	Complex	87	Symbolic data handling statement
7441	Real part	871	Algebraic expression manipulation statement
7442	Imaginary	872	List handling statement
745	Formal (= algebraic)	876	String handling statement
746	String	878	Pattern handling statement
747	List of pointer	8A	Operating system (and/or equipment)
748	Hierarchical	8A0Z	Input-output system
74B	Combinations of variable and constant	8A1	Input statement
76	File description		
78	Format description		
781	Input		
7812	F notation/floating point/real type		
7813	I notation/integer/fixed point type		

8A10Z	<i>By Media</i>	8AG	Storage and segmentation
8A11	Control console		allocation statement
8A12	Punched card	8AJ	Operating system and
8A13	Paper tape		machine feature
8A14	Magnetic tape		
8A15	Magnetic disc	8AJ0Z	<i>By Machine feature</i>
8A16	Magnetic cards	8AJ1	Single address
8A17	Printed magnetic character	8AJ2	Two address
8A18	Sketch pad and light pen	8AJ3	Three address
8A1A	Electrical signal	8AJ4	Four address
8A1B	Printed/typed/hand written	8AJ8	Multiple address
	letter and numeral	8AJ813	One or three address
8A1C	Voice	8AJA	Addressing
8A5	Output statement	8AJA1	Direct addressing
		8AJA2	Indirect addressing
8A50Z	<i>By Media</i>	8AJF	Register/accumulator
8A51	Control console	8AJF1	Console typewriter
8A52	Punched card	8AJF2	Remote console
8A53	Paper tape	8AJH	Pointer access
8A54	Magnetic tape	8AJJ	Multi access
8A55	Magnetic disc	8AJM	Special features
8A56	Magnetic card	8C	Functional statement
8A57	Printed copy	8E	Complex assignment state- ment
8A58	Cathode ray tube		
8A5A	Electric signal	8E1	Real part
8A5C	Voice	8E2	Imaginary part
8AC	Library reference statement/ Macro statement	8G	Presumptive instruction 1 unmodified instruction
8AD	Debugging statement		

8 Examples

81 NOTE

Some of the documents selected as example deal with multi-focal subjects. But in the examples given in Sec 83, only one or two of the subjects have been selected from each of such documents for the purpose of demonstration of the construction of Class Number using the depth schedule given in Sec 7.

82 ALPHABETICAL INDEX TO SUBJECTS

Given below is an alphabetical index to the subjects of the documents listed in Sec 83 Classified Part. The Serial Number of the entry in Sec 83 is given as the Index Number against each entry in this section. The alphabetical subject index has been prepared according to Chain Indexing procedure. As all the subjects of the documents given in the examples deal with Programming Language, this Basic Subject Term is mentioned in the Subject Index Entry only when it is necessary to get a correct interpretation of the subject heading.

Algebraic differentiation, Meta-semantic language, ALGOL-like	10
ALGOL 60	14
ALGOL-like	10
Algorithmic language, ALGOL 60	14

- Application
- Computer design, Information retrieval, (for) Binary computer, Machine independent, LOGOL 19
 - Graph processing
 - Storage tube, GPD, Programming language 17
 - Subset of ALGOL 11
 - Information retrieval, (for) Binary computer, Machine independent, LOGOL 19
 - Meta-semantic language, ALGOL-like 10
 - Operations research, Graph processing, Subset of ALGOL 60 11
 - Problem-oriented language, PDEL 22
 - Research, Graph processing, Storage tube, GPD 17
 - Subset of COBOL 12
 - Arithmetic expression, Algebraic differentiation, Meta-semantic language, ALGOL-like 10
 - Assembly language
 - (for) 360 computer, ALGOL-like, PL 360 21
 - Machine independent 9
- Backus Naur form, Problem-oriented, User-oriented, Programming language 2
- Basic element, Operations research application, Graph processing, Subset of ALGOL 60 11
- Closure, Compiling language 5
- COBOL 15
- Compiler, Graph processing, Translating language 8
- Compiling language 4-5
- Complete grammar, Reference language, GEDANKEN 16
- Complex arithmetic, Semantic table, Arithmetic expression, Algebraic differentiation, Meta-semantic language, ALGOL-like 10
- Compound data structure, Experimental, Typeless, Complete grammar, Reference language, GEDANKEN 16
- Computer used,
 - ALGOL-like, PL360 21
 - LOGOL 18
 - Programming language 7
- Construction, Identifier, General purpose, COBOL 15
- Control, Extension of PL1 13
- Data
 - description
 - Experimental, Typeless, Complete grammar, Reference language, GEDANKEN 16
 - Semantic table, Arithmetic expression, Top-down syntax analyser, Algebraic differentiation, Meta-semantic language, ALGOL-like 10
 - manipulation, Phrase structure grammar, Symbolic machine language, (for) 360 computer, ALGOL-like, PL360 21
 - Declaration and non-executable statement
 - Complete grammar, Reference language, GEDANKEN 16
 - Semantic table, Arithmetic expression, Algebraic differentiation, Meta-semantic language, ALGOL-like 10
 - Display, GPD 17
- Ease of conversion, Programming language 8
- Equipment, GPD 17

- Experimental, Typeless, Complete grammar, Reference language, GEDANKEN 10
- Extension of PL1 13
- Function
 - Backus Naur form, Problem-oriented, User-oriented language 2
 - Extension of PL1 13
 - GPDL 17
 - Low level language 3
 - Machine independent, LOGOL 19
 - Phrase structure grammar, Symbolic machine language, Assembly language, (for) 360 computer, ALGOL-like, PL360 21
 - Problem-oriented language, (for) Special purpose digital computer 7
 - Procedure-oriented, Non-specialist language, SPARTA 23
 - Storage tube, GPDL 17
 - Subset of ALGOL 11
 - User-oriented, Non-procedural language, UL1 24
- Functional
 - characteristic
 - Algebraic differentiation, Meta-semantic language, ALGOL-like 10
 - Complete grammar, Reference language, GEDANKEN 16
 - Macro statement, One-or-two address, Computer design application, Information retrieval, Machine independent, (for) Binary computer, LOGOL 19
 - form
 - Compiling language 4
 - General purpose
 - COBOL 15
 - PL1 20
 - Operations research application, Graph processing, Subset of ALGOL 60 11
 - Functional
 - part
 - Algorithmic language, ALGOL 60 14
 - Experimental, Typeless, Complete grammar, Reference language, GEDANKEN 16
- GEDANKEN, Programming language 16
- General purpose
 - COBOL, Programming language 15
 - Control, Extension of PL1 13
 - PL1 19
 - GPDL 17
- Grammar
 - Problem-oriented, User-oriented language 2
 - Reference language, GEDANKEN 16
 - Symbolic machine language, Assembly language, (for) 360 computer, ALGOL-like, PL360 21
- Graph processing
 - Backus Naur form, Problem-oriented, User-oriented language 2
 - Procedure-oriented, Non-specialist language, SPARTA 23
 - Storage tube, GPDL, Programming language 17
 - Subset of ALGOL 60 11
 - Translating to ALGOL 8
- Greek alphabet, Input language, (for) Digital computer 6
- Identifier
 - Compiling language 4

- General purpose
 - COBOL 15
 - PL1 20
- Information processing
 - (for) Binary computer, Machine independence, LOGOL 19
 - Low level language 3
 - Problem-oriented language, (for) Special purpose digital computer 7
 - User-oriented, Non-procedural language, UL1 24
- Information retrieval
 - By Boolean query, Low level language 3
 - Problem-oriented language, (for) Special purpose digital computer 7
 - Machine independent, LOGOL 19
 - User-oriented, Non-procedural language, UL1 24
- Input language, Digital computer 6
- JOVIAL 18
- Language L-like 10
 - PL360 21
- Linguistic character
 - Data manipulation, Phrase structure grammar, Symbolic machine language, Assembly language, (for) 360 computer, ALGOL-like, PL360 21
 - Graph processing, Backus Naur form, Problem-oriented, User-oriented language 2
 - Problem-oriented language 1
- LOGOL 19
- Low level language 3
- Machine
 - feature, Computer design application, Information retrieval, (for) Binary computer, Machine independent, LOGOL 19
 - independent
 - LOGOL 19
 - Programming language 9
- Macro statement, (for) One-or-two address, Computer design application, Information retrieval, (for) Binary computer, Machine independent, LOGOL 19
- Maintenance, Identifier, General purpose language, PL1 20
- Mathematical model, Problem-oriented language, JOVIAL 18
- Meta-semantic language, ALGOL-like 10
- Node, Operations research application, Graph processing, Subset of ALGOL 60 11
- Non-procedural language, UL1 23
- Non-specialist language, SPARTA 22
- Operating system, (for) One-or-two address, Computer design application, Information retrieval, (for) Binary computer, Machine independent, LOGOL 19
- PDEL 22
 - Phrase structure grammar, Symbolic machine language, Assembly language, (for) 360 computer, ALGOL-like, PL360 21
 - PL1 20
 - PL360 21
- Problem-oriented language 1
 - (for) Special purpose digital computer 7
- JOVIAL 18

- FDEL 22
 - User-oriented 2
 - Procedure-oriented, Non-specialist language, SPARTA 23
 - Programming language 1-24
 - Property, Macro-statement, (for) One-or-two address, Computer design application, Information retrieval, (for) Binary computer, Machine independent, LOGOL 19
 - Pseudo-assembly, Machine independent language 9
- Reference language, GEDANKEN 16
- Semantics
 - Graph processing, Backus Naur form, Problem-oriented, User-oriented language 2
 - Table, Arithmetic expression, Algebraic differentiation, Meta-semantic language, ALGOL-like 10
- Soft-ware development
 - (for) Digital computer, Programming language 6
 - Graph processing, Translating language 8
 - Pseudo-assembly, Machine independent language 9
- Solution, Differential equation, Problem-oriented language, FDEL 22
- SPARTA 23
- Storage tube, GFDL 17
- Subset
 - ALGOL 60 11
 - COBOL 12
- Symbol manipulation
 - Backus Naur form, Problem-oriented, User-oriented language 2
 - Procedure-oriented, non specialist language, SPARTA 23
 - Translating to ALGOL 8
- Symbolic machine language, Assembly language (for) 360 Computer, ALGOL-like, PL360 21
- Syntax
 - Data manipulation, Phrase structure grammar, Symbolic machine language, Assembly language, (for) 360 computer, ALGOL-like, PL 360 21
 - Problem-oriented language 1
- Teaching, Subset of COBOL 12
- Technical characteristic
 - Graph processing, Backus Naur form, Problem-oriented, User-oriented language 2
 - Problem-oriented language 1
 - Translating to ALGOL 8
 - Triplex, Algorithmic, ALGOL 60 14
- Type of compatibility 9-13
 - PL 360 20
- Typeless, Complete grammar, Reference language, GEDANKEN 16
- User
 - oriented language 2
 - Non-procedural language, UL1 24
 - SPARTA 23
 - written language 2
 - Non-procedural, UL1 24
 - SPARTA 23
- Variable, Input language, (for) Digital computer 6

Writing property, Macro statement, (for) One-or-two address, Computer design application, Information retrieval, (for) Binary computer, Machine independent, LOGOL 19

83 Classified Part

- 9S COMPUTER SCIENCE
9S, 8 Programming Language
9S, 8-LH,093
- 1 N69 VESSLOV (Y M) and KOPROV (V M). Algorithms and algorithmic language (AD-702909;1969;32). (Comp abstr., 14;1970;192);
9S,8-LN-LH-G5-CR8;094
USER-ORIENTED, PROBLEM-ORIENTED LANGUAGE, BACKUS
- 2 N67 SCHWYNN (P M). Problem-oriented graphic languages. (ACM Nat Cong (22). Proceedings. 1967. P 471-7). (Comp abstr. 12;1968;173).
9S,8-LU1-CT4B2
PROGRAMMING LANGUAGE, LOW LEVEL LANGUAGE, INFORMATION RETRIEVAL, BY BOOLEAN QUERY
- 3 N68 EVANS (D) and DANI (A VAN). Data structure programming system. (Inf Proc. 1968;557-64). (Comp abstr. 14;1970;121).
9S,8-P6;162
COMPILING LANGUAGE, IDENTIFIER
- 4 N68 WHITNEY (G E). Generation and recognition properties of table languages. (Inf proc. 1968;388-94). (Comp abstr. 14;1970;122).
9S,8-96;631
COMPILING LANGUAGE, CLOSURE
- 5 N68 WHITNEY (G E). Generation and recognition properties of table languages. (Inf proc. 1968;388-94). (Comp abstr. 14;1970;122).
9S,8-R0(B1-(M112))-C811;41113
PROGRAMMING LANGUAGE, (FOR) DIGITAL COMPUTER, INPUT LANGUAGE; GREEK ALPHABET.
- 6 N70 KUZMENKO (G E) and SEMIK (V P). Input language and address translator for the digital computer Minsk 12. (AD-703784; 70;14). (Comp abstr. 14;1970;195).
9S,8-R0(B1-ZA1)-LH-CT4
PROGRAMMING LANGUAGE, (FOR) SPECIAL PURPOSE DIGITAL COMPUTER, PROBLEM-ORIENTED LANGUAGE, INFORMATION RETRIEVAL
- 7 N66 TURKOV (V V). Use of a query language for synthesis of digital computer. (Eng cyber. 1966;64-70). (Comp abstr. 14;1970;11).
9S,8-R25AL-CR8-C82
PROGRAMMING LANGUAGE, TRANSLATION TO ALGOL, GRAPH PROCESSING, COMPILER

- 8 N68 CRESPIREGHIZZI (S) and MORPURGO (F). Graph theory oriented extension to ALGOL. (Calcolo. 5;1968;451-93). (Comp abstr. 14;1970;148).
9S,8-R32-P42-C8
PROGRAMMING LANGUAGE, MACHINE INDEPENDENT, PSEUDO-ASSEMBLY LANGUAGE, SOFTWARE DEVELOPMENT
- 9 N70 ADVANCED PROGRAMMING technique for a cost-effective hardware independent realization of naval software system. (Comp Symb Inc. AD-702058;1970;52). (Comp abstr. 14;1970;170).
9S,8-R36AL-LS-C9(B33,1)-94B1-91F;744
ALGOL-LIKE, META-SEMANTIC LANGUAGE, ALGEBRAIC DIFFERENTIATION, ARITHMETIC EXPRESSION, SEMANTIC TABLE, COMPLEX
- 10 N68 FOXLEY (F) and KING (P). Meta-semantic language for use with a top down syntactic analyser. (Inf proc. 1968;366-72), (Comp abstr. 14;1970;122).
9S,8-R37AL60-CR8-C9 (BTT);14221
SUBSET OF ALGOL 60, GRAPH PROCESSING, APPLICATION TO OPERATIONS RESEARCH, NODE
- 11 N70 CRESPIREGHIZZI (S) and MORPURGO (R). Language for treating graphs. (CACM. 13;1970;319-23). (Comp abstr. 14;1970;148).
9S,8-R37C-C9(T:3)
SUBSET OF COBOL, TEACHING
- 12 N69 GILES (P). Mini COBAL. (Comp j. 12;1969;208-14).
9S,8-R38PL1-CF-C2
EXTENSION OF PL 1, CONTROL, GENERAL PURPOSE
- 13 N69 BOULTON (P I P) and REID (P A). Process control language. (IEEE Trans comp. C-18;1969;1049-53).
9S,8-(AL-60)-LH-9623
ALGOL 60, Algorithmic language, Triplex
- 14 N68 KULSCH (J) and others. Algorithmic language Triplex—ALGOL 60. (Numerische mathematika. 11;1968;175-80). (Comp abstr. 12;1968;172).
9S,8-(C)-C2;162:36
COBOL, GENERAL PURPOSE, IDENTIFIER, CONSTRUCTION
- 15 N69 BARNETT (M R). Programming language support of data bases. (File org. 1969;331-45). (Comp abstr. 14;1970;37).
9S,8-(GE)-LP-GY-9710-1;622
GEDANKEN, REFERENCE LANGUAGE, COMPLETE GRAMMAR, TYPELESS, EXPERIMENTAL, COMPOUND DATA STRUCTURE
- 16 N70 REYNOLDS (J C). GEDANKEN—simple typeless language based on the principles of completeness and reference concept. (CACM;13;1970;308-19). (Comp abstr. 14;1970;147).
9S,8-(GPDJ)-CV37-CR8-C9(A:f)
GPDJ, STORAGE TUBE, GRAPH PROCESSING, FOR RESEARCH.

- 17 N70 NOTLEY (M G). Graphical picture drawing language. (Compul. 40;1970;68-74).
9S,8-(J)-LH;y7(B)
JOVIAL, PROBLEM-ORIENTED LANGUAGE, MATHEMATICAL MODEL
- 18 N69 GROSSURO (S). Algorithmic languages. (System Dev Corp. AD-702812;1969;13). (Comp abstr. 14;1970;192).
9S,(L)-R32-R0(B1-zd2)-CT4-C9(D8B1-b2)-98AJ82-98AC;071
LOGOL, Machine independence, (for) Binary computer. Information retrieval, Application to computer design. One-or-two address, Macro statement, Writing property.
- 19 N68 HUCL (C). LOGOL and its application to computer development. (Inf. proc. 1968;479-93) (Comp abstr. 14;1970;122).
9S,8-(PL1)-C2:162:82
PL 1, GENERAL PURPOSE, IDENTIFIERS, MAINTENANCE
- 20 N69 BARNETT (M R). Programming language support of data base. (File org. 1964;331-45). (Comp abstr. 14;1970;37).
9S,8-(PL360)-R36AL-R0(B1-(360))-P4-IG-GP-CD;093
PL360-ALGOL LIKE, (FOR) 360 COMPUTER, ASSEMBLY LANGUAGE, SYMBOLIC MACHINE LANGUAGE, PHRASE STRUCTURE GRAMMAR, DATA MANIPULATION, SYNTAX
- 21 N68 WIRTH (Niklaus). PL360, a programming language for the 360 computer. (J ACM;15;1968;31-74).
9S,8-(PDEL)-LH-C9(B33,4)
PDEL-PROBLEM-ORIENTED LANGUAGE, SOLUTIONS OF DIFFERENTIAL EQUATION
- 22 N70 CHARDENS (A F) and KARPLUS (W J). PDEL-a language for Partial differential equations. (c ACM;13;1970;184-91).
9S,8-(SPA)-LN5-LL-CR8
SPARTA, NON-SPECIALIST, PROCEDURE-ORIENTED LANGUAGE, GRAPH PROCESSING
- 23 N68 MEZET (C). SPARTA, a procedure-oriented programming language for the manipulation of arbitrary line drawings. (Inf. proc. 1968;597-604). (Comp abstr. 14;1970;124).
9S,8-(UL1)-LT-LN-CT4
UL 1, NON-PROCEDURAL LANGUAGE, USER-ORIENTED LANGUAGE, INFORMATION RETRIEVAL
- 24 N68 OLLE (T W). UL 1: a non-procedural language for retrieving information from data bases. (Infor proc. 1968;572-8). (Comp abstr. 14;1970;124).

91 Acknowledgement

We are thankful to Prof A Neelameghan for guiding us in the work reported in this paper.

92 Bibliographical References

- 1 Sec 343 ANJANEYULU (V). Computer production engineering: Depth classification. (Annual seminar (DRTC). 5;1967; Paper H).
- 2 Sec 41 NEELAMEGHAN (A) and GOPINATH (M A). Grouping of quasi isolates. (Annual seminar (DRTC). 4;1966; Paper K).
- 3 Sec 321 PYLYSHYN (Z W). *Ed.* Perspectives on the computer revolution. 1970. P 61.
- 4 Sec 1 RANGANATHAN (S R). Design of depth classification: Methodology. (Lib sc. 1;1964; Paper A).
- 5 Sec 1 —. Prolegomena to library classification. Ed 3. Assist by M A Gopinath. 1967.
- 6 Sec 322 SMITH (T M). Some perspectives on the early history of computers. [*In* Pylyshyn (Z W), *Ed.*. Perspectives on the computer revolution. 1970. P 8].