

Principles of Programming

Section 12: Contents, Reference Data, and Index

IBM Personal Study Program

Contents

Contents of Sections 1 to 11.....	1
IBM 1401 Instructions with Symbolic Programming System Mnemonics	3
Autocoder Operation Codes	7
Card and Computer Character Codes.....	11
Instruction Timing Data	12
Machine Configuration Assumed in Text.....	13
Glossary	14
Bibliography	22
Answers to Selected Exercises.....	27
Index	54

Section 12: Contents, Reference Data, and Index

Contents of Sections 1 to 11

1	THE NATURE OF DATA PROCESSING	
1.1	Introduction	1
1.2	Basic Data Processing Ideas	2
1.3	An Example of Sequential File Processing	6
1.4	An Example of Random Access File Processing	20
	Exercises	24
2	INTRODUCTION TO COMPUTING EQUIPMENT	
2.1	The IBM Punched Card	1
2.2	IBM 1401 Data Processing System Components	5
2.3	The Card Sorter and Collator	15
2.4	System Components Used in Sequential File Processing	18
2.5	Representation of Information in a Computer	20
	Exercises	24
3	CODING FUNDAMENTALS	
3.1	Computer Storage and Its Addressing	1
3.2	Instructions	5
3.3	Storage of Instructions	13
3.4	Arithmetic and Control Registers	15
3.5	Addition and Subtraction	20
	Exercises	23
4	SYMBOLIC PROGRAMMING	
4.1	Fundamentals of Symbolic Programming	1
4.2	Further Information on the SPS Language and Processor	11
4.3	Case Study: Payroll	14
	Exercises	24
5	BRANCHING	
5.1	Fundamentals of Branching	1
5.2	Further Branching Operations	8
5.3	Case Study: Parts Explosion and Summary	15
	Exercises	23
6	ADDRESS MODIFICATION AND LOOPS	
6.1	Computations on Addresses	1
6.2	Program Switches	5

6.3	Program Loops	7
6.4	Address Modification Loops	9
6.5	Indexing	19
	Exercises	25
7	MISCELLANEOUS OPERATIONS	
7.1	Editing and Format Design	1
7.2	Printer Carriage Control	6
7.3	Input and Output Timing	10
7.4	Buffering	16
7.5	Program Timing	18
7.6	Subroutines and Utility Programs	20
	Exercises	23
8	MAGNETIC TAPE OPERATIONS	
8.1	Physical Characteristics of Magnetic Tapes	1
8.2	Magnetic Tape Instructions	6
8.3	Tape Programming with Autocoder and IOCS	14
8.4	Inventory Control Case Study	23
	Exercises	31
9	RANDOM ACCESS FILE STORAGE	
9.1	Basic Concepts	1
9.2	The IBM 1405 and 1301 Disk Storage Units	2
9.3	Disk Storage Programming for the IBM 1405	5
9.4	Disk Organization and Addressing	15
9.5	Disk Storage Utility Routines	18
9.6	Case Study: Wholesale Grocery	20
	Exercises	25
10	PLANNING AND INSTALLING A COMPUTER APPLICATION	
10.1	Problem Statement	1
10.2	Problem Analysis	3
10.3	Block Diagram and Program for Inventory Control Processing	7
10.4	Program Checkout	13
10.5	Going into Operation	16
10.6	Documentation	17
10.7	Summary	18
	Exercises	19
11	ADDITIONAL PROGRAMMING METHODS	
11.1	Introduction	1
11.2	Decision Tables	1
11.3	The FORTRAN Coding System	4
11.4	The Report Program Generator	7
11.5	The COBOL Programming System	12
11.6	Fundamentals of COBOL Programming	13
11.7	COBOL Program for Inventory Control Case Study	19

IBM 1401 Instructions with Symbolic Programming System Mnemonics

Not all of the instructions listed here have been discussed in the text. For those which have been introduced, the section and page number of the detailed description is given.

Where no d-modifier is shown, none is required; where it is shown as "d", consult the appropriate table below for meanings.

Instructions involving special features not found on a basic 1401 are marked with *.

Input-Output Instructions

Instruction	Actual Op Code — d	SPS Op Code — d	Section and Page
Read a Card	1	R	3-8
Write a Line	2	W	3-9
Write Word Marks	2	W	3-9
Write and Read	3	WR	7-15
Punch a Card	4	P	3-9
*Read Punch Feed	4	R P	R
Read and Punch	5	RP	7-15
Write and Punch	6	WP	7-15
*Write and Read Punch Feed	6	R WP	R
Write, Read, and Punch	7	WRP	7-15
*Start Read Feed	8	SRF	
*Start Punch Feed	9	SPF	

Arithmetic Instructions

Instruction	Actual Op Code — d	SPS Op Code — d	Section and Page
Add	A	A	3-20
Subtract	S	S	3-22
Zero and Add	?	ZA	6-10
Zero and Subtract	!	ZS	6-10
*Multiply	@	M	4-18
*Divide	%	D	

Logic Instructions

Instruction	Actual Op Code — d	SPS Op Code — d	Section and Page
Branch	B	B	5-1
Branch If Indicator On	B d	B d	5-2
Branch If Character Equal (Contents of B address compared with d-modifier.)	B d	B d	8-27
Branch If Word Mark and/or Zone Compare	V C	BWZ C	5-8 5-9

d-modifier for Branch If Indicator On Instruction

	blank	Unconditional
COMPARISON	/	Unequal compare: $B \neq A$
	S	*Equal compare: $B = A$
	T	*Low compare: $B < A$
	U	*High compare: $B > A$
OVERFLOW	Z	Overflow
LAST CARD	A	Last card switch
SENSE SWITCHES	B	*Sense switch B
	C	*Sense switch C
	D	*Sense switch D
	E	*Sense switch E
	F	*Sense switch F
	G	*Sense switch G
PRINTER	@	Carriage channel 12
	9	Carriage channel 9
	P	*Printer busy
	R	*Carriage busy
I/O ERRORS	?	Reader error with I/O check stop switch off
	!	Punch error with I/O check stop switch off
	≠	Printer error with I/O check stop switch off
PROCESSOR ERROR	%	Processing error with process check stop switch off

MAGNETIC TAPE

K *End of reel
L *Tape error

DISK FILE

V *Read/Write parity check or read back check error
W *Wrong-length record
X *Unequal address compare
Y *Any disk storage error
N *Access inoperable

Data Control Instructions

Instruction	Actual Op Code — d	SPS Op Code — d	Section and Page
Move Characters to A or B		MCW	3-7
Word Mark <i>read control to PR IN M</i>		LCA	4-15
Load Characters to A Word Mark <i>L read cont</i>			
Move Characters and Suppress <i>(Zeros -) of print</i>	Z	MCS	5-4
Move Characters and Edit	E	MCE	4-20
Move Numerical	D	MN	
Move Zone	Y	MZ	
Set Word Mark	,	SW	3-7
Clear Word Mark	□	CW	3-8

Miscellaneous Instructions

Instruction	Actual Op Code — d	SPS Op Code — d	Section and Page
Control Carriage	F d	CC d	7-8
Select Stacker	K d	SS d	5-20
No Operation	N	NOP	6-6
Clear Storage	/	CS	3-10
Halt	.	H	5-7
*Store A-address Register	Q	SAR	
*Store B-address Register	H	SBR	
*Modify Address	#	MA	

7-8

MANUAL

Magnetic Tape Instructions

A-address is of the form %Ux where x is the tape unit number.

Instruction	Actual Op Code	d	SPS Op Code	d	Section and Page
Read Tape	M	R	MCW	R	8-8
Write Tape	M	W	MCW	W	8-7
Read Tape with Word Marks	L	R	LCA	R	
Write Tape with Word Marks	L	W	LCA	W	
Control Unit	U	d	CU	d	8-7
*Move Characters to Record or Group Mark	P		MCM		
*Move and Insert Zeros	X		MIZ		

d-modifiers for Control Unit Instruction

B	Backspace Tape Record
E	Skip and Blank Tape
M	Write Tape Mark
R	Rewind Tape
U	Rewind Tape and Unload

Disk Storage Instructions

A-address is of the form %FX where:

- X = 0 is used with operation code M for Seek Disk
- X = 1 specifies single record
- X = 2 specifies full track
- X = 3 is used with operation code M for Write Disk Check

Instruction	Actual Op Code	d	SPS Op Code	d	Section and Page
Seek Disk	M	R	MCW	R	9-6
Read Disk	M	R	MCW	R	9-7
Write Disk	M	W	MCW	W	9-8
Read Disk with Word Marks	L	R	LCA	R	
Write Disk with Word Marks	L	W	LCA	W	

Autocoder Operation Codes

DECLARATIVE OPERATIONS

Type	Mnemonic Op Code	Description	Actual Op Code	d
	DA	Define Area		
	DC	Define Constant (No Word Mark)		
	DCW	Define Constant with Word Mark		
	DS	Define Symbol		
	DSA	Define Symbol Address		
	EQU	Equate		

IMPERATIVE OPERATIONS

Arithmetic	A	Add	A	
	D	Divide	%	
	M	Multiply	@	
	S	Subtract	S	
	ZA	Zero and Add	?	
	ZS	Zero and Subtract	!	
Data Control	MBC	Move and Binary Code	M	B
	MBD	Move and Binary Decode	M	A
	MCE	Move Characters and Edit	E	
	MCS	Move Characters and Suppress Zeros	Z	
	MIZ	Move and Insert Zeros	X	
	MLC	} Move Characters to Word Mark	M	
	MCW			
	MLCWA	} Move Characters and Word Marks to Word Mark in A-Field	L	
	LCA			
	MLNS	} Move Single Numerical Character	D	
	MN			
	MLZS	} Move Single Zone	Y	
	MZ			
MRCM	} Move Characters to Record Mark or Group Mark — Word Mark	P		
MCM				

Type	Mnemonic Op Code	Description	Actual Op Code — d		
Logic	B	Branch Unconditional	B		
	BAV	Branch on Arithmetic Overflow	B	Z	
	BBE	*Branch if Bit Equal	W	d	
	BC9	Branch on Carriage Channel 9	B	9	
	BCV	Branch on Carriage Overflow (12)	B	@	
	BE	Branch on Equal Compare (B = A)	B	S	
	BEF	Branch on End of File or End of Reel	B	K	
	BER	Branch on Tape Transmission Error	B	L	
	BH	Branch on High Compare (B > A)	B	U	
	BIN	*Branch on Indicator	B	d	
	BL	Branch on Low Compare (B < A)	B	T	
	BLC	Branch on Last Card (Sense Switch A)	B	A	
	BM	Branch on Minus (11-zone)	V	K	
	BPCB	Branch Printer Carriage Busy	B	R	
	BPB	Branch Printer Busy	B	P	
	BU	Branch on Unequal Compare (B ≠ A)	B	/	
	BW	Branch on Word Mark	V	1	
	BWZ	*Branch on Word Mark or Zone	V	d	
	BCE	*Branch if Character Equal	B	d	
	BSS	*Branch if Sense Switch On	B	A-G	
	C	Compare	C		
	I/O Commands	BSP	Backspace Tape	U	B
		CU	*Control Unit	U	d
DCR		Disengage Character Reader	U	D	
ECR		Engage Character Reader	U	E	
LU		*Load Unit	L	d	
MU		*Move Unit	M	d	
P		Punch	4		
PCB		Punch Column Binary	4	C	
R		Read	1		

* d-character must be coded in the operand of the instruction.

Type	Mnemonic Op Code	Description	Actual Op Code — d	
	RCB	Read Column Binary	1	C
	RD	Read Disk Single Record	M	R
	RDT	Read Disk Full Track	M	R
	RDW	Read Disk Single Record with Word Marks	L	R
	RDTW	Read Disk Full Track with Word Marks	L	R
	RF	Read Punch Feed	4	R
	RP	Read and Punch	5	
	RT	Read Tape	M	R
	RTB	Read Tape Binary	M	R
	RTW	Read Tape with Word Marks	L	W
	RWD	Rewind Tape	U	R
	RWU	Rewind and Unload Tape	U	U
	SD	Seek Disk	M	
	SKP	Skip and Blank Tape	U	E
	SPF	Start Punch Feed	9	
	SRF	Start Read Feed	8	
	W	Write	2	
	WD	Write Disk Single Record	M	W
	WDC	Write Disk Check	M	
	WDCW	Write Disk Check with Word Marks	L	
	WDT	Write Disk Full Track	M	W
	WDTW	Write Disk Full Track with Word Marks	L	W
	WDW	Write Disk Single Record with Word Marks	L	W
	WM	Write Word Marks	2	W
	WP	Write and Punch	6	
	WR	Write and Read	3	
	WRF	Write and Read Punch Feed	5	R
	WRP	Write, Read and Punch	7	
	WT	Write Tape	M	W
	WTB	Write Tape Binary	M	W
	WTM	Write Tape Mark	U	M
	WTW	Write Tape with Word Marks	L	W

Type	Mnemonic Op Code	Description	Actual Op Code — d
Miscellaneous	CC	*Carriage Control	F d
	CCB	*Carriage Control and Branch	F d
	CS	Clear Storage	/
	CW	Clear Word Mark	□
	H	Halt	.
	MA	Modify Address	#
	NOP	No Operation	N
	SAR	Store A-Address Register	Q
	SBR	Store B-Address Register	H
	SS	*Select Stacker	K 1, 2, 4, 8
SSB	*Select Stacker and Branch	K 1, 2, 4, 8	
SW	Set Word Mark	,	

* d-character must be coded in the operand of the instruction.

CONTROL OPERATIONS

Type	Mnemonic Op Code	Description
	CTL	Control
	END	End
	ENT	Enter New Coding Mode
	EX	Execute
	LTORG	Literal Origin
	ORG	Origin

PRINTS AS	DEFINED CHARACTER	CARD CODE	BCD CODE
	BLANK		
.	.	12-3-8	BA8 21
□	□	12-4-8	CBA84
(Left Parenthesis (Special Character)	12-5-8	BA84 1
<	Less Than (Special Character)	12-6-8	BA842
#	Group Mark (Note 1)	12-7-8	CBA8421
&	&	12	CBA
\$	\$	11-3-8	CB 8 21
*	*	11-4-8	B 84
)	Right Parenthesis (Special Char.)	11-5-8	CB 84 1
;	Semicolon (Special Character)	11-6-8	CB 842
Δ	Delta (Mode Change)	11-7-8	B 8421
—	—	11	B
/	/	0-1	C A 1
/	/	0-3-8	C A8 21
%	%	0-4-8	AB4
=	Word Separator	0-5-8	C AB4 1
'	Apostrophe (Special Character)	0-6-8	C AB42
⌈	Tape Segment Mark	0-7-8	AB421
¢	Cent (Special Character Note 2)		A
+	+	3-8	8 21
@	@	4-8	C 84
:	Colon (Special Character)	5-8	84 1
>	Greater Than (Special Character)	6-8	842
✓	Tape Mark	7-8	C 8421
&	? (Plus Zero)	12-0	CBA8 2
A	A	12-1	BA 1
B	B	12-2	BA 2
C	C	12-3	CBA 21
D	D	12-4	BA 4
E	E	12-5	CBA 4 1
F	F	12-6	CBA 42
G	G	12-7	BA 421
H	H	12-8	BA8
I	I	12-9	CBA8 1
-	(Minus Zero)	11-0	B 8 2
J	J	11-1	CB 1
K	K	11-2	CB 2
L	L	11-3	B 21
M	M	11-4	CB 4
N	N	11-5	B 4 1
O	O	11-6	B 42
P	P	11-7	CB 421
Q	Q	11-8	CB 8
R	R	11-9	B 8 1
#	# Record Mark	0-2-8	AB 2
S	S	0-2	C A 2
T	T	0-3	A 21
U	U	0-4	C A 4
V	V	0-5	A 4 1
W	W	0-6	A 42
X	X	0-7	C A 421
Y	Y	0-8	C AB
Z	Z	0-9	AB 1
0	0	0	C 8 2
1	1	1	1
2	2	2	2
3	3	3	C 21
4	4	4	4
5	5	5	C 4 1
6	6	6	C 42
7	7	7	421
8	8	8	8
9	9	9	C 8 1

LOW

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

BA

12

10

11

01

0 (+A)

00

NOVE (+A)

H1

The 1401 has the ability to read MLP card codes in the read feed only. The 1401 ignores the 8-9 punches when they appear in the same column. The 1401 does not punch out MLP card codes.

Note 1. If specified, this code can be made compatible with 705 Group Mark Code (12-5-8).

Note 2. The A-bit coding must be program generated in the 1401 (it cannot be read from a card; it can be punched as a zero). It is used in conjunction with the C-bit to indicate a blank position on tape that was written in even-bit parity.

Glossary

Terminology in the computing field is not yet fully standardized. Every attempt has been made in this book to use terms in their most common meaning, but it should be realized that variations do exist. This listing, therefore, is not intended to be complete or rigorous; it is intended simply to provide a basic vocabulary.

ABSOLUTE CODING — coding in which instructions are written in the basic machine language, that is, with absolute addresses and actual operation codes.

ACCUMULATOR — a storage register where results are accumulated.

ADDITION RECORD — a record that results in the creation of a new record in a master file being updated.

ADDRESS — a label, name or number which designates a register, a location or a device where information is stored; the part of an instruction which specifies the location of an operand.

ADDRESS COMPUTATION — computer operations which result in the creation or modification of the address parts of instructions.

ALPHAMERIC CHARACTERS — characters which may be either letters of the alphabet, numerical digits, or certain special symbols.

ANALOG COMPUTER — a computer which represents variables by physical analogies in continuous form, such as amount of rotation of a shaft, amount of voltage, etc. Contrasted to digital computer; the difference is sometimes expressed by saying that an analog computer *measures* whereas a digital computer *counts*.

ARITHMETIC UNIT — that component of a computer where arithmetic and logical operations are performed.

ASSEMBLE — to translate a routine coded in a symbolic machine language into absolute machine instructions and to assign machine storage for those instructions and for data; usually done by the computer under control of an assembly routine. Distinguished from *compile* by the fact that assembly produces one machine instruction from one symbolic instruction, whereas compiling produces (in general) *many* machine instructions from one pseudo instruction.

BATCH PROCESSING — the system of processing in which a number of similar input items are grouped for processing during the same machine run.

BINARY DIGIT — one of the symbols 0 or 1. A digit in the binary scale of notation, usually called a *bit*.

BLANK — the character which results in storage from reading an input record such as a card column which contains no punches; the character code in storage which will result in not printing in a given position.

BLOCK — a group of records, words or characters handled as one unit. Used in this book primarily to denote a group of records on magnetic tape.

BLOCK DIAGRAM — a graphic representation of the logical sequence of procedural steps for processing data. More detailed than a flow chart; a flow chart shows the overall steps to be performed, whereas a block diagram shows the details of *how* to perform each step.

BLOCKING — combining two or more records into one block; usually refers to tape operations.

BRANCH — a point in a routine where one of two or more choices is selected under control of the routine, that is, a conditional transfer.

BUFFER STORAGE — any device which temporarily stores information during a transfer of information. From a programming standpoint, refers to a device for matching the speeds of internal computation and an input or output device, thereby permitting simultaneous computation and input or output.

CARD FIELD — a fixed number of consecutive card columns assigned to a unit of information.

CELL — see Location.

CHAINING — (1) 1401 instruction addresses: a technique of omitting one or both addresses of an instruction with the omitted address being supplied by the previous contents of the corresponding address register; (2) disk storage: a system of storing records in a disk file in which each record belongs to a chain (group of records) and has a linking field for tracing the chain.

CHARACTER — one of a set of elementary symbols which may be arranged in ordered groups to express information; these symbols may include the decimal digits 0 through 9, the letters A through Z, punctuation symbols, special input and output symbols, and any other symbols which a computer may accept.

CHECKOUT — the process of determining the correctness of a computer routine, locating any errors in it, and correcting them. Also the detection and correction of malfunction in the computer itself.

CLOSED SUBROUTINE — a subroutine not stored in the main path of the routine. Such a subroutine is entered by a branch operation and provision is made to return control to the main routine at the end of the subroutine.

CODE — to write instructions for a computer either in absolute or some other language.

COLLATE — to merge items from two or more similarly sequenced files into one sequenced file without necessarily including all items from the original files.

COLLATING SEQUENCE — the sequence into which the allowable characters of a particular computer are ranked.

COMPARE — to examine the representation of two groups of characters to discover identity or relative magnitude.

COMPILE — to produce a machine-language routine by translation from a program written in some non-machine language. See also Assemble.

COMPILER — a special machine-language routine used to perform compiling operations.

COMPLEMENT — usually represents the negative of a quantity. For example, the three-digit ten's complement of 026 is 974.

COMPUTER — any device capable of accepting information, processing it and providing the results of the processing in acceptable form. In this text the term is always meant to imply a *stored program digital* computer.

CONSOLE — a part of the computer where most of the external controls for a computer operation are exercised and where most of the indicators of internal operation are located.

CONTROL CARD — a card which contains input data or parameters for a specific application of a general routine.

CONTROL FIELD — the field of information by which a record in a file is identified and/or controlled.

CONTROL PANEL — a panel which uses removable wires to direct the operation of some computers and of punched card equipment. Not used in the 1401.

CONTROL TOTAL — the sum of the amounts in a particular field in an arbitrary grouping of records; may have some significance as a number; used for checking machine, program and data reliability.

CONTROL UNIT — that portion of the hardware of the computer which directs the sequence of automatic operations, interprets the coded instructions, and initiates the proper signals to the computer circuits to execute the instructions.

CORE STORAGE — a form of high speed storage in which information is represented by the magnetization of ferromagnetic cores.

DATA PROCESSING — a generic term for all of the operations carried out on data according to precise rules of procedure; a generic term for computing in general as applied to business situations.

DEBUGGING — see Checkout.

DELETION RECORD — a record which results in the deletion of some corresponding records from a master file.

DETAIL FILE — a file to be processed against a master file.

DIGITAL COMPUTER — a computer in which information is represented in discrete form, such as by one of two directions of magnetization of a magnetic core, or by the presence or absence of an electric pulse at a certain point in time. Contrasted with analog computer.

DOCUMENT — any representation of information which is readable by human beings; usually on paper.

EDIT — to rearrange information for machine output or input. To prepare for publication — that is, delete, rearrange, select, or insert data as needed.

EXECUTE — to carry out an instruction or perform a routine.

FIELD — a set of one or more characters treated as a whole; a unit of information.

FILE — a collection of records; an organized collection of information directed toward some purpose.

FILE MAINTENANCE — the processing of a master file to handle changes in it. Examples: changes in number of dependents in a payroll file, the addition of new checking accounts in a banking application.

FIXED WORD LENGTH COMPUTER — one in which a computer word always contains the same number of characters. Contrasted with variable word length.

FLOW CHART — a graphic representation of the sequence of operations required to carry out a data processing procedure. More general than a block diagram; a flow chart shows the sequence of processing steps, whereas a block diagram shows in detail how to carry out each step.

FORM — a printed or typed document which usually has blank spaces for the insertion of information.

FORMAT — the predetermined arrangement of characters, fields, lines, page number, punctuation marks, etc. Refers to input, output and file information.

GENERATE — to produce a complete routine from one which is in skeleton form under control of parameters supplied to the generator routine.

HARDWARE — the mechanical, magnetic, electric and electronic devices from which a computer is constructed.

HASH TOTAL — a control total that has no meaning in itself as a number.

HEADER LABEL — a magnetic tape block at the beginning of a tape, which identifies and describes the information on the tape.

HOME RECORD — the first record in a chain of records, using the chaining method of disk file organization.

HOUSEKEEPING — operations in a routine which do not directly contribute to the solution of the problem at hand, but which are made necessary by the method of operation of the computer. Examples: Loop testing, setting of word marks.

INDEX REGISTER — a register which contains a quantity that may be used to automatically modify addresses (and for other purposes) under direction of the control section of the computer.

INITIALIZE — to execute the instructions immediately prior to a loop, which set addresses, counters, data, etc., to their desired initial values. See also Loop.

INPUT — information transferred from auxiliary or external storage into the internal storage of a computer.

INSTRUCTION — a set of characters which as a unit causes the computer to perform one of its operations. An instruction may contain one or more addresses according to the number of references to operands in storage contained in the instruction.

INTERNAL STORAGE — computer storage for data and instructions, from which instructions can be moved directly to the control unit for execution.

INTERPRET — (1) to print on a punched card the information punched in that card; (2) to translate non-machine language to machine language.

INTERPRETIVE ROUTINE — a routine which decodes instructions written in non-machine language and immediately executes those instructions. Contrasted with a *compiler*, which decodes the non-machine language and produces a machine-language routine to be executed at a later time.

KEY — see Control Field.

LABEL — (1) in SPS programming, the symbolic location of a word; (2) in magnetic tape operations, a record magnetically recorded on a tape to identify its contents to a computer routine.

LIBRARY — an organized collection of standard and proven routines and subroutines which may be incorporated in larger routines.

LINKAGE — a technique for providing interconnections between a main routine and a closed subroutine.

LOCATION — a place in storage where a unit of data or of an instruction may be stored.

LOOP — a coding technique whereby a group of instructions is repeated with modification of some of the instructions within the group and/or with modification of the data being operated upon. Usually consists of initialization, computing modification and testing, although not necessarily in that order.

MACHINE LANGUAGE — a language for writing instructions in a form to be executed directly by the computer. Contrasted to symbolic coding languages and to procedure-oriented languages.

MACRO-INSTRUCTION — a machine-like source-language statement which can produce a number of machine instructions when compiled.

MAGNETIC DISK — a storage device in which information is recorded on the magnetizable surface of a rotating disk. A magnetic disk storage system is an array of such devices with associated reading and writing heads which are mounted on movable arms.

MAGNETIC DRUM — a storage device in which information is recorded on the magnetizable surface of a rotating cylinder.

MAGNETIC TAPE — a storage system in which information is recorded on the magnetizable surface of a strip of plastic tape.

MASTER FILE — a file of semipermanent reference information which is usually updated periodically.

MEMORY — see Internal Storage.

MERGE — to combine items from two or more similarly sequenced files into one sequenced file, including all items from the original files.

MICROSECOND — one millionth of a second.

MILLISECOND — one thousandth of a second.

MNEMONIC OPERATION CODE — an operation code written in a symbolic notation that is easier to remember than the actual operation code of the machine. Must be converted to an actual operation code before execution which is done as part of an assembly, interpretive or compiling routine.

OBJECT ROUTINE — the machine-language routine which is the output after translation from the source language. The running routine.

OFF-LINE — pertaining to the operation of input or output devices or auxiliary equipment not under direct control of the central processing unit.

ON-LINE — pertaining to the operation of input or output devices under direct control of the computer.

OPEN SUBROUTINE — a subroutine which is inserted directly into a larger routine where needed.

ORIGIN — the absolute storage address of the beginning of a program.

OUTPUT — information transferred from the internal storage of a computer to output devices or external storage.

OVERFLOW — (1) the generation of a quantity beyond the capacity of a register; (2) a record linked to a home record, in the chaining method of disk file organization.

PARAMETER — a quantity to which arbitrary values may be assigned; used in subroutines and generators specifying such things as record size, decimal point location, record format, etc.

PARITY CHECK — a checking technique based on making the total number of 1's in some grouping of binary digits odd (or even). Whenever such a group is read, it is presumed to be correct if the number of 1's is still odd (or even).

PROCEDURE-ORIENTED LANGUAGE — a source language oriented to the description of procedural steps in machine computing.

PROCESSOR — a program of instructions that carries out the translation from a source-language program to an object program. Includes compilers, assemblers, report program generators, etc.

PROGRAM (VERB) — to plan the method of attack on a specified and defined problem for computer solution. Distinguished from coding by the fact that coding involves writing instructions whereas programming is characterized by the drawing of flow charts.

PROGRAM (NOUN) — a group of related routines which solve a given problem.

PSEUDO INSTRUCTION — a symbolic representation of information to an assembler or a compiler; not an instruction to the computer, although for convenience it is often written in the same general format as a computer instruction.

RANDOM ACCESS STORAGE — storage in which the time required to obtain information is relatively independent of the location of the information most recently obtained.

READ — to transfer information from an input device to internal storage.

REAL TIME COMPUTATION — a data processing arrangement in which the computer is required to be able to supply information to a physical or business activity whenever the information is demanded.

RECORD — a collection of fields; the information related to one area of activity in a data processing activity; files are made up of records.

REGISTER — a device that can hold information while or until it is used. May consist of core storage.

REPORT GENERATION — a technique for producing complete machine reports from information which describes the input file and the format and contents of the output report.

REWIND — to return a tape to its beginning.

ROUTINE — a set of computer instructions that carries out some well defined function.

RUN — one routine or several routines automatically linked so that they form an operating unit during which manual interruptions are not normally required of the computer operator.

SOFTWARE — all the programming systems required for an effective data processing operation, in addition to the hardware of the computer system itself. Includes assemblers, compilers, utility routines, etc.

SOURCE LANGUAGE — the language used to specify computer processing; translated into object language by an assembler or compiler.

STORAGE — any device into which information can be transferred, which will hold information and from which the information can be obtained at a later time.

STORED PROGRAM COMPUTER — a computer which can alter its own instructions in storage as though they were data and later execute the altered instructions.

SUBROUTINE — a routine which may be incorporated into a larger routine.

SWITCH — a symbol used to indicate a branch point or a set of instructions to condition a branch for later execution.

SYMBOLIC CODING — coding in which instructions are written in non-machine language. That is, coding using symbolic notation for operators, operands, and locations instead of actual machine instruction codes and addresses.

SYSTEMS ANALYSIS — the analysis of a business activity to determine precisely what must be accomplished and how to accomplish it.

TRAILER LABEL BLOCK — a block which follows one or more other blocks and contains data pertinent to the preceding blocks.

TRANSACTION FILE — a file containing current information related to a data processing activity; usually used to update a master file.

UPDATE — to modify a master file according to current information, often that contained in a transaction file, according to a procedure specified as part of a data processing activity.

UTILITY ROUTINE — a standard routine used to assist in the operation of a computer. For instance, a conversion routine, a print out routine, a tape reading routine, etc.

VARIABLE WORD LENGTH COMPUTER — one in which the number of characters comprising a computer word is not fixed.

WORD — a set of characters having one addressable location and treated as one unit.

WORKING STORAGE — a portion of internal storage used for input data, intermediate results, or output.

WRITE — to transfer information from internal storage to an output device or to auxiliary storage.

ZERO ELIMINATION — the process of eliminating non-significant zeros to the left of significant digits usually before printing.

Bibliography

Many good books on computing are available, and some of them are listed here. A more complete listing may be found in the *IBM Data Processing Bibliography* (J20-8014-2).

Gotlieb, C. C., and J. N. P. Hume: *High Speed Data Processing*. McGraw-Hill Book Co., New York, 1958, 11 + 338 pp.

Begins with an introduction to data processing and programming ideas, followed by a number of chapters on typical commercial applications.

Grabbe, Eugene M., editor: *Automation in Business and Industry*. John Wiley and Sons, Inc., New York, 1957, 611 pp.

Contains information on a wide variety of computer applications as of 1956.

Grabbe, Eugene M., Simon Ramo, and Dean E. Wooldridge, editors: *Handbook of Automation, Computation, and Control*. Vol. 2: Computers and Data Processing. John Wiley and Sons, Inc., New York, 1959, 1100 pp.

Contains a long chapter on the theory of programming, several chapters on typical applications, and long sections on computer design and analog computers.

Gregory, Robert H., and Richard L. VanHorn: *Automatic Data-Processing Systems: Principles and Procedures*. Wadsworth Publishing Company, San Francisco, 1960, 705 pp.

Introduces concepts in data processing, computing equipment, programming, and systems design. The coverage is thorough and thoughtful.

Hein, Leonard W.: *Introduction to Electronic Data Processing for Business*. D. Van Nostrand Company, Inc., Princeton, 1961, 14 + 320 pp.

A general introduction to computer programming, based on the IBM 650. All of the examples are based on commercial applications. Includes chapters on file maintenance, merging and collating, sorting, and report writing.

Kaufman, Felix: *Electronic Data Processing and Auditing*. The Ronald Press Co., New York, 1961, 180 pp.

The title of this book is somewhat misleading. Although the emphasis is on auditing problems, there is a great deal of valuable general information on applications and systems design. Contains many flow charts and block diagrams, many examples of the flow of information through a business organization, and discussions of breaking an application into runs, as well as the treatment of questions of error-checking and proving accuracy.

Leeds, Herbert D., and Gerald M. Weinberg: *Computer Programming Fundamentals*. McGraw-Hill Book Co., New York, 1961, 368 pp.

Designed for both individual and classroom training in basic computer programming. The IBM 7090 is used as the illustrative computer, but the understanding of its functions and purpose may easily be adapted to any computer course. Heavy emphasis is laid on the basic concepts, ideas and techniques required for the purpose of communication with computers.

Martin, E. W., Jr.: *Electronic Data Processing, An Introduction*. Richard D. Irwin, Inc., Homewood, Illinois, 1961, 423 pp.

Ranges from punched card methods to management problems in introducing and using a computer. The programming section is based on the IBM 650. Contains much valuable information on applications and systems design.

McCormick, E. M.: *Digital Computer Primer*. McGraw-Hill Book Co., New York, 1959, 214 pp.

Discusses the components, operation, and organization of a digital computer. Does not discuss programming or applications, except incidentally. The treatment is elementary, requiring little mathematics.

McCracken, D. D.: *A Guide to FORTRAN Programming*. John Wiley and Sons, Inc., New York, 1961, 88 pp.

An introduction to the FORTRAN system, requiring little mathematics.

McCracken, D. D., H. Weiss, and T. H. Lee: *Programming Business Computers*. John Wiley and Sons, Inc., New York, 1959, 510 pp.

Discusses computer coding and programming in terms of a hypothetical computer quite different from the IBM 1401. Includes chapters on tape programming methods, program checkout procedures, sorting methods, etc.

Murphy, J. S.: *Basics of Digital Computers*. John F. Rider Publishers, New York, 1958. Vol. I: 116 pp.; Vol. II: 133 pp.; Vol. III: 136 pp.

An extremely readable and profusely illustrated introduction to computer design. Introduces basic programming concepts, but in no depth. For a quick picture of how a computer works, this is a good book.

Phister, Montgomery, Jr.: *Logical Design of Digital Computers*. John Wiley and Sons, Inc., New York, 1958, 408 pp.

A standard engineering text on the design and operation of computers.

Richards, R. K.: *Arithmetic Operations in Digital Computers*. D. Van Nostrand Co., Inc., Princeton, 1955, 397 pp.

Clear explanation of the design and workings of a computer. Considerably more technical than Murphy, but less so than Phister.

Wrubel, Marshall H.: *A Primer of Programming for Digital Computers*. McGraw-Hill Book Co., New York, 1959, 230 pp.

A general introduction to computer programming, based on the IBM 650 and several programming systems for it. Many of the examples involve scientific applications, although the mathematical preparation required is not extensive. Discussions of symbolic and automatic coding techniques are included.

IBM Publications

These and other IBM publications are available from local IBM branch offices.

1401 Data Processing System. General Information Manual, form D24-1401-2.

A brief introduction to the 1401, with a sketch of some basic computer concepts. Not a complete description of the machine, and not intended to teach programming.

1401 Data Processing System. Reference Manual, form A-24-1403-4.

A complete description of the programming and operating characteristics of the 1401. Contains few examples; not intended as a teaching text.

IBM 1401 Data Processing System. From Control Panel to Stored Program. General Information Manual, form F20-0208.

A brief introduction to the 1401 and to programming concepts, followed by a description of how punched card operations can be implemented on the 1401. Good introduction to programming for the person with punched card experience.

Introduction to IBM Data Processing Systems. General Information Manual, form F22-6517.

A manual prepared to provide a basic understanding of computer systems and programmed functions. Used as an introductory text in IBM education programs.

Flow Charting and Block Diagramming Techniques. Reference Manual, form C20-8008-0.

Discusses the need for flow charting and block diagramming, introduces the accepted IBM notation, and gives a number of examples.

IBM Charting and Diagramming Template. Form X24-5884-5.

Simplifies the drawing of flow charts and block diagrams, as well as providing a convenient source of various reference information on printing and card punching.

IBM 1401 Symbolic Programming System: Preliminary Specifications. Bulletin, form J29-0200-2.

IBM Magnetic Tape Record Characteristics. Form X22-6785.

Report Program Generator for IBM 1401 Card and Tape Systems. Bulletin, form J24-0215.

Autocoder for the IBM 1401: Preliminary Specifications. Bulletin, form J24-1434-1.

Utility Programs for IBM 1401 Tape Systems: Preliminary Specifications. Bulletin, form J29-1411-0.

Utility Programs for IBM RAMAC 1401 Systems: Preliminary Specifications. Bulletin, form J29-1426-0.

COBOL. General Information Manual, form F28-8053-1.

A brief introduction to data processing and the COBOL language, followed by a complete definition of the language and several examples.

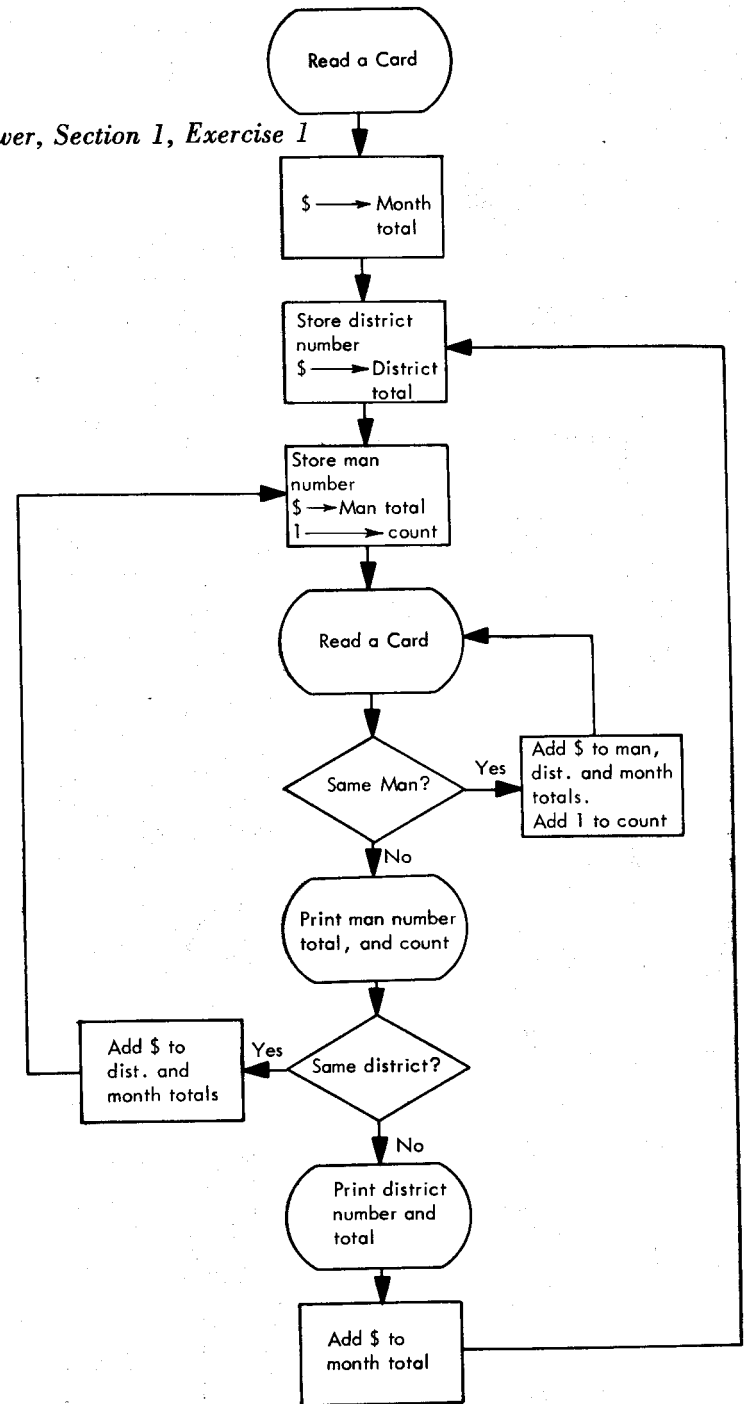
An Introduction to Information Retrieval. General Information Manual, form E20-8044.

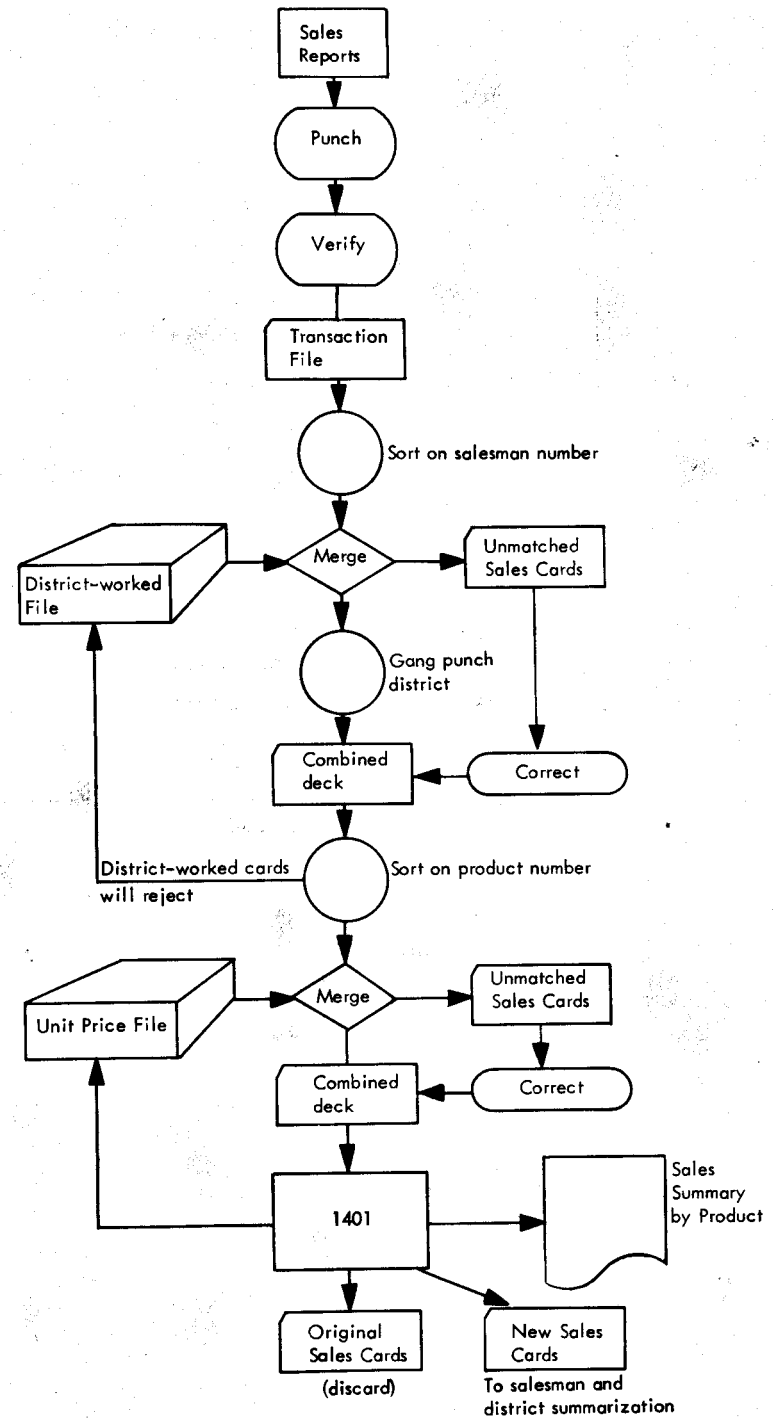
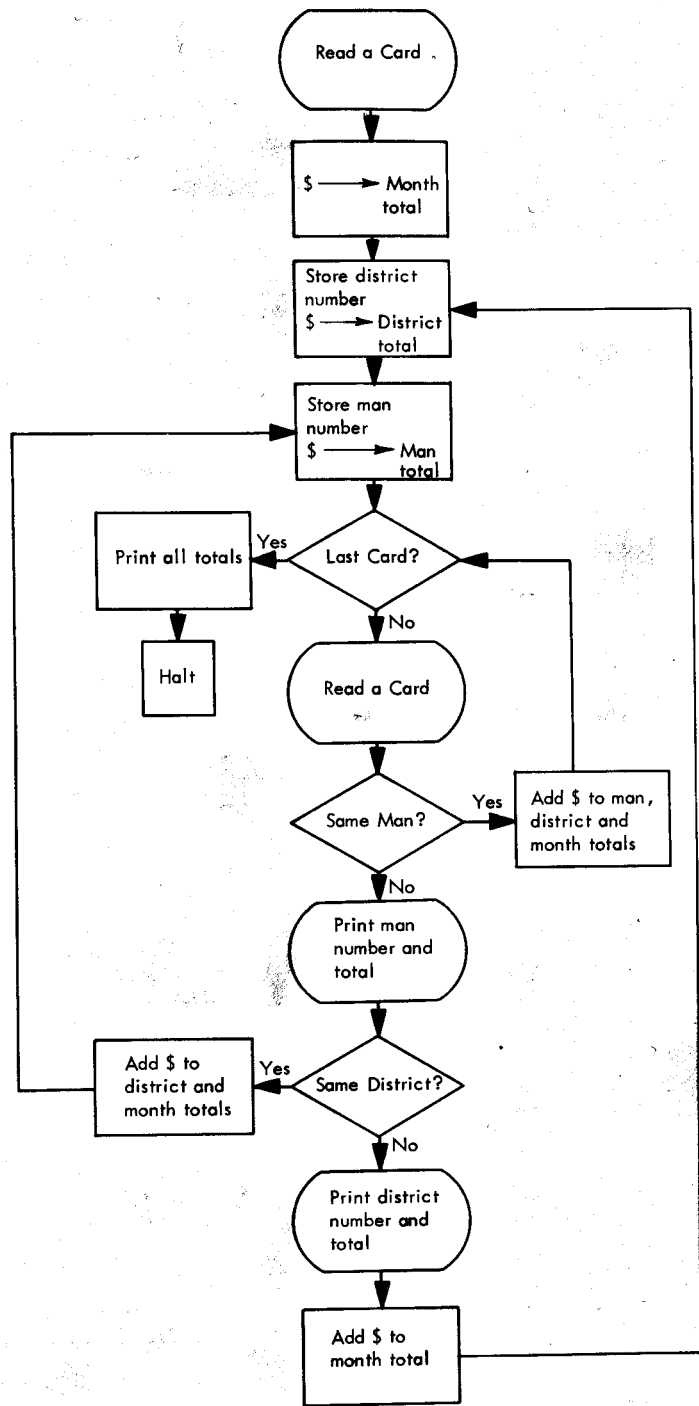
FORTRAN. General Information Manual, form F28-8074

IBM Data Processing Bibliography. Form J20-8014-2.

Answers to Selected Exercises

Answer, Section 1, Exercise 1





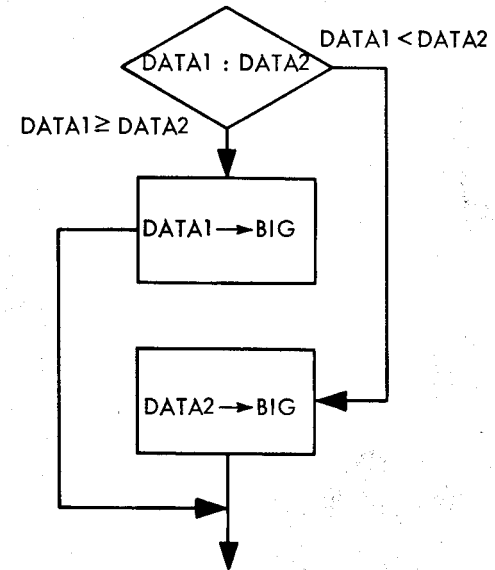
Answer, Section 4, Exercise 1

START	0333
REPEAT	0349
RØUND	0400
TØTAL	0411
READ1	0010
PRINT1	0209

Answer, Section 4, Exercise 3

PG	LIN	CT	LABEL	OP	A OPERAND	B OPERAND	D	LOC	INSTRUCTION
1	010			ORG	0500				
1	020	4	ABC	CS	0080			0500	/ 080
1	030	4		CS	0299			0504	/ 299
1	040	4		CS	0332			0508	/ 332
1	050	7		SW	A1 -004	A2 -004		0512	, 001 006
1	060	7		SW	A3 -004	B1 -003		0519	, 011 207
1	070	1	BCD	R				0526	1
1	080	7		A	A1	TOT		0527	A 005 571
1	090	7		A	A2	TOT		0534	A 010 571
1	100	7		S	A3	TOT		0541	S 015 571
1	110	7		A	HALFD	TOT -001		0548	A 572 570
1	120	7		MCS	TOT -002	B1		0555	Z 569 210
1	130	4		W	BCD			0562	2 526
1	140		A1	DS	0005			0005	
1	150		A2	DS	0010			0010	
1	160		A3	DS	0015			0015	
1	170		B1	DS	0210			0210	
1	180	6	TOT	DCW	*		000000	0571	
1	190	1	HALFD	DCW	*		5	0572	
1	200			END	ABC				/ 500 080

Answer, Section 5, Exercise 1



IBM INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
 CODING SHEET

Program _____
 Programmed by _____ Date _____

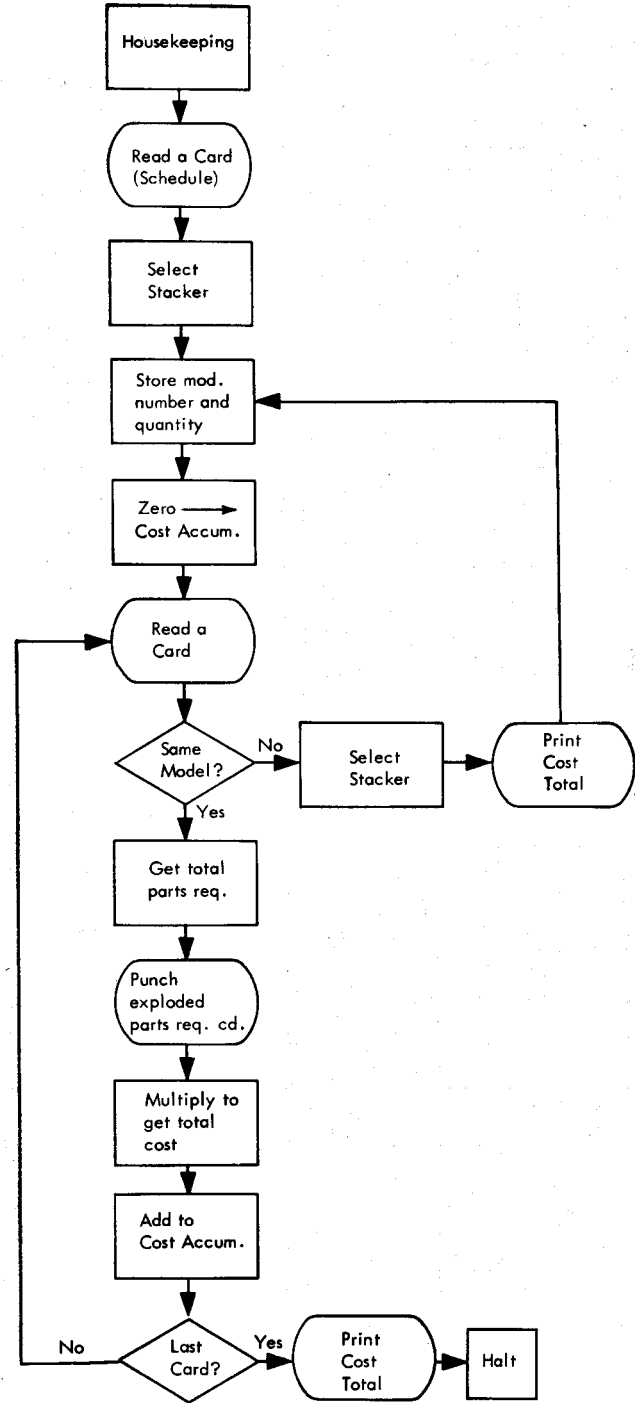
LINE	COUNT	LABEL	OPERATION	(A) OPERAND				(B) OPERAND				d			
				ADDRESS	±	CHAR. ADJ.	IND	ADDRESS	±	CHAR. ADJ.	IND				
3	5	6	7	8	13	14	16	17	23	27	28	34	36	38	40
0 1 0			C	D A T A 1						D A T A 2					
0 2 0			B	Ø T H E R											U
0 3 0			M C W	D A T A 1						B I G					
0 4 0			B	A R Ø U N D											
0 5 0			Ø T H E R	M C W D A T A 2						B I G					
0 6 0			A R Ø U N D												
0 7 0															

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

Page No. 2 of
Identification 76 80

Programmed by _____ Date _____

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS
				ADDRESS	±	CHAR. ADJ.	ADDRESS	±	CHAR. ADJ.	
3	5, 6, 7, 8	13	14	16, 17	23	27, 28	29	34	38, 39, 40	55
0.1.0			CS	0.0.8.0						
0.2.0			CS	0.1.8.0						
0.3.0			CS	0.2.9.9						
0.4.0			CS	0.3.3.2						
0.5.0			SW	0.0.4.1						
0.6.0			R							
0.7.0			BWZ	PUNCH			0.0.2.3			
0.8.0			MCW	0.0.8.0			0.2.4.0			
0.9.0			W	AROUND						
1.0.0		PUNCH	MCW	0.0.8.0			0.1.4.0			
1.1.0			P							
1.2.0		AROUND								
3.0										



INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
 CODING SHEET

IBM

FORM 324-1152-2
 PRINTED IN U.S.A.

Program _____ Date _____
 Programmed by _____ Identification $\frac{76}{80}$ of _____

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS							
				ADDRESS	CHAR. ADJ.	±	ADDRESS	CHAR. ADJ.	±								
3	5	7	8	13	14	16	17	23	24	27	28	34	35	38	39	40	
0.1.0		EXPSUM	C.S.	0180													HOUSEKEEPING
0.2.0			C.S.	0180													
0.3.0			C.S.	0299													
0.4.0			C.S.	0332													
0.5.0			S.W.	0001								0006					
0.6.0			S.W.	0011								0031					
0.7.0			S.W.	0034								0131					
0.8.0			R														
0.9.0			S.S.														
1.0.0		SCHCD	LCA R I														READ SCHED CARD
1.1.0			MCWR2														SELECT STACKER
1.2.0			MCWZER0														PRODUCT NUMBER
1.3.0		PARTCD	C														QUANTITY SCHED.
1.4.0			R I														ZERO COST ACCUM.
1.5.0			B PRINT														SAME PROD NO.
1.6.0			LCA R 3														NO-NEW SCH CARD
1.7.0			LCA R 4														YES-PART CARD
1.8.0			MCWQTY														SETUP PUNCH AREA
1.9.0			M R 5														GET TOTAL QTY
2.0.0			MCWMULTI														
2.1.0			P														PUNCH EXP. PT. REQ

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
 CODING SHEET

IBM

FORM 324-1152-2
 PRINTED IN U.S.A.

Program _____ Date _____
 Programmed by _____ Identification $\frac{76}{80}$ of _____

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS							
				ADDRESS	CHAR. ADJ.	±	ADDRESS	CHAR. ADJ.	±								
3	5	7	8	13	14	16	17	23	24	27	28	34	35	38	39	40	
0.1.0			MCWQTY														SETUP COST MULT.
0.2.0			M R 6														
0.3.0			A MULT2														
0.4.0			B LAST														
0.5.0			PARTCD														
0.6.0		PRINT	S.S.														
0.7.0			LCA PRDND0														
0.8.0			LCA QTY														
0.9.0			LCA EDIT														
1.0.0			MC ECSTACC														
1.1.0			W SCHCD														
1.2.0		LAST	LCA PRDND0														
1.3.0			LCA QTY														
1.4.0			LCA EDIT														
1.5.0			MC ECSTACC														
1.6.0			W														
1.7.0		HALT	H														
1.8.0																	
1.9.0																	
2.0.0																	

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

FORM 224-1152-2
PRINTED IN U.S.A.

Program _____ Date _____
Page No. 1 of 2 Identification 76 80

LINE	COUNT	LABEL	OPERATION	(A1) OPERAND		(B1) OPERAND		COMMENTS
				ADDRESS	CHAR ADJ.	ADDRESS	CHAR ADJ.	
0.1.1.0	R.1		D.S.	0.0.0.5				
0.2.0	R.2		D.S.	0.0.0.9				
0.3.0	R.3		D.S.	0.0.1.0				
0.4.0	R.4		D.S.	0.0.3.0				
0.5.0	R.5		D.S.	0.0.3.3				
0.6.0	R.6		D.S.	0.0.3.8				
0.7.0	P.R.O.D.N.O		D.S.	0.1.0.5				
0.8.0	P.A.R.T.N.O		D.S.	0.1.1.0				
0.9.0	D.E.S.C		D.S.	0.1.3.0				
1.0.0	0.5	T.O.T.Q.T.Y	D.C.W.	0.1.3.5				
1.1.0	0.8	M.U.L.T.I	D.C.W.*					
1.2.0	1.0	M.U.L.T.2	D.C.W.*					
1.3.0	0.7	C.S.T.A.C.C	D.C.W.*					
1.4.0	0.5	P.R.I.N.T.1	D.C.W.	0.2.0.5				
1.5.0	0.4	P.R.I.N.T.2	D.C.W.	0.2.1.3				
1.6.0	0.8	P.R.I.N.T.3	D.C.W.	0.2.2.5				
1.7.0	0.7	Z.E.R.O	D.C.W.*	0.0.0.0	0.0.0.0			
1.8.0	0.4	Q.T.Y	D.C.W.*		0.0.0.0			
1.9.0	0.8	E.D.I.T	D.C.W.*		0.0.0.0			
2.0.0			E.N.D.E.X.P.S.U.M.					

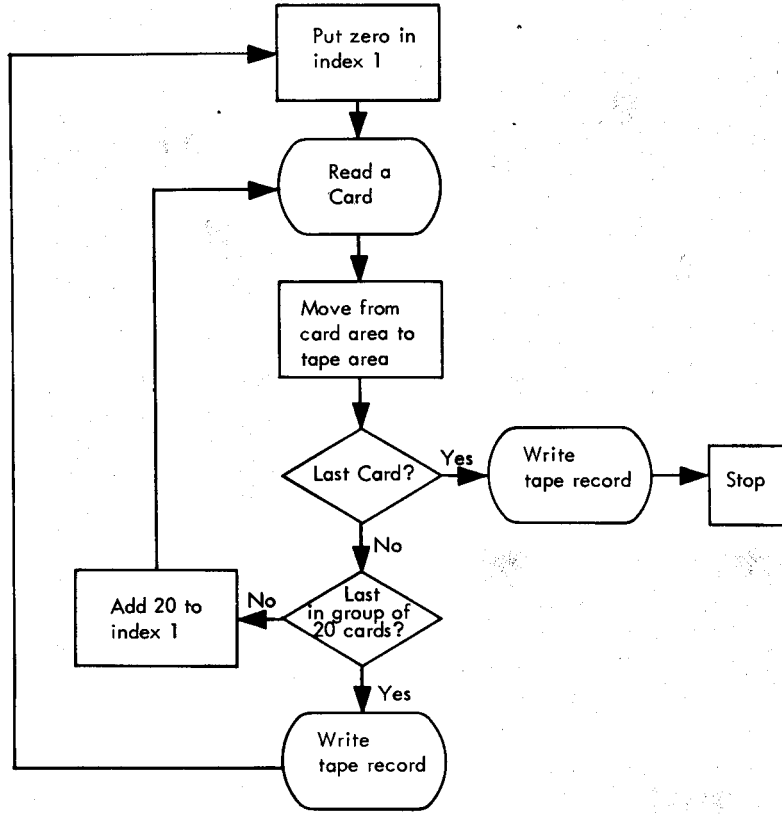
IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

FORM 224-1152-2
PRINTED IN U.S.A.

Program _____ Date _____
Page No. 1 of 2 Identification 76 80

LINE	COUNT	LABEL	OPERATION	(A1) OPERAND		(B1) OPERAND		COMMENTS
				ADDRESS	CHAR ADJ.	ADDRESS	CHAR ADJ.	
0.1.1.0	R.1		S.T.A.R.T.					
0.2.0	Z.A		Z.E.R.O					
0.3.0	M.C.W		I.N.I.T.A.D.D					
0.4.0	A.D.D.I.N.S.A		O.O.O					
0.5.0	A		F.I.V.E					
0.6.0	C		A.D.D.I.N.S.+0.0.3					
0.7.0	B		A.D.D.I.N.S					
0.8.0	M		C.I					
0.9.0	A		F.I.V.E					
1.0.0	M.C.S		A.C.C.U.M.-0.0.6					
1.1.0	M.C.S		O.O.O.8					
1.2.0	W		L.A.S.T					
1.3.0	B		S.T.A.R.T					
1.4.0	B		S.T.A.R.T					
1.5.0	H		M					
1.5.0	1.4	A.C.C.U.M	D.C.W.*					
1.7.0	0.1	Z.E.R.O	D.C.W.*					
1.8.0	0.3	I.N.T.A.D.D	D.C.W.*					
1.8.0	0.1	F.E.S.T	D.C.W.*					
2.0.0	0.1	F.I.V.E	D.C.W.*					
0.6.C.I			D.C.W.*	0.8.3.3.3				
			E.N.D.S.T.A.R.T					



FORM X24-1132-2
PRINTED IN U.S.A.

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

IBM

Program _____

Programmed by _____

Date _____

Page No. 1 of 2

Identification 76 - 80

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS
				ADDRESS	CHAR. ADJ.	±	ADDRESS	CHAR. ADJ.	±	
3	5 6 7 8			27 28	29	30	31 32	33 34	35	
0 1 0		START	SW	0 0 0 1					SET CARD AREA W/M	
0 2 0		ZERIND	MCWZERØ						INITIALIZE INDEX	
0 3 0		READ								
0 4 0			MCW0 0 2 0						INDEXED MOVE	
0 5 0			WRITE 1						LAST CARD 0	
0 6 0			C INDEX 1							
0 7 0			B WRITE 2						NO LAST IN GROUP	
0 8 0			A TWENTY						YES	
0 9 0			B READ						NO MODIFY INDEX	
1 0 0		WRITE 1	<i>write tape record</i>							
1 1 0		WRITE 2	H # - 0 0 3						FINAL STOP	
1 2 0		WRITE 2	<i>write tape record</i>							
1 3 0			ZERIND							
1 4 0	0 3	ZERØ	DCW #							
1 5 0	0 3	INDEX 1	DCW 0 0 8 9							
1 6 0	0 3	TEST	DCW # 3 8 0							
1 7 0	0 2	TWENTY	DCW # 2 0							
1 8 0			END START							
1 9 0										
2 0 0										

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

Page No. 1 of 2
Identification 78

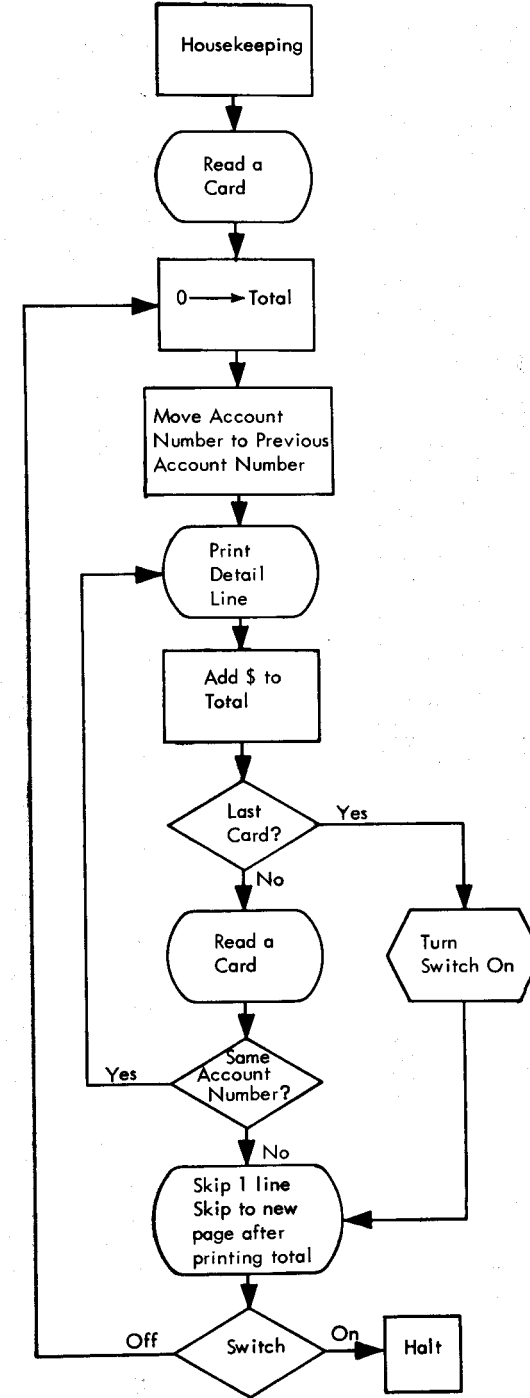
Date _____

IBM

Program _____

Programmed by _____

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS
				ADDRESS	±	CHAR. ADJ.	ADDRESS	±	CHAR. ADJ.	
3	5 6 7 8									55
0.1	0		MCS	A			0.2	0		
0.2	0		MCS	S			0.2	1	4	
0.3	0		SW	D			0.2	1	9	
0.4	0		MCS	D			0.2	1	9	
0.5	0		MCS	D			0.2	2	2	
0.6	0		MCS	D			0.2	2	4	
0.7	0		CW	D			0.2	3	6	
0.8	0		LCA	E			0.2	3	6	
0.9	0		MCE	D			0.2	3	6	
1.0	0									
1.1	0									
1.2	0									
1.3	0	09	E	D			0			
1.4	0									



FORM 274-1152-2
PRINTED IN U.S.A.

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

IBM

Program _____ Date _____
 Programmed by _____ Identification 76 of 80

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS
				ADDRESS	±	CHAR. ADJ.	ADDRESS	±	CHAR. ADJ.	
3	5 6 7 8	13 14	15 16 17	23	27	28	34	39 40	85	
0.1.0		S.T.A.R.T.	S.W.	O.O.O.I		O.O.2.3			H.O.U.S.E,K.E.E.P.I.N.G.	
0.2.0			R.							
0.3.0		A.	M.C.W.Z.E.R.O.S.			T.O.T.A.L.			I.N.I.T.I.A.L.I.Z.E.A.C.C.U.M	
0.4.0		B.	M.C.W.O.O.O.5			P.R.A.C.N.O.			P.R.E.V.A.C.C.T.N.O.	
0.5.0			M.C.S.O.O.O.5			O.2.O.5			A.C.C.T.N.O.	
0.6.0			L.C.A.E.D.I.T.			O.2.I.6			S.E.T.U.P.E.D.I.T.	
0.7.0			M.C.E.O.O.2.8			O.2.I.6				
0.8.0			W.							
0.9.0			A.	O.O.2.8		T.O.T.A.L.			A.C.C.U.M.A.M.O.U.N.T.	
1.0.0			B.	C.					A.L.A.S.T.C.A.R.D.O.	
1.1.0			R.						N.O.	
1.2.0			C.	O.O.O.5		P.R.A.C.N.O.			S.A.M.E.A.C.C.T.N.O..O	
1.3.0			B.	B.					S.Y.E.S	
1.4.0		D.	C.C.						J.N.O.-S.P.A.C.E.I.L.I.N.E	
1.5.0			L.C.A.E.D.I.T.			O.2.I.6			S.E.T.U.P.E.D.I.T.	
1.6.0			M.C.E.T.O.T.A.L.			O.2.I.6			A.S.E.I.U.P.S.K.I.P.	
1.7.0			C.C.						P.R.I.N.T.A.N.D.S.K.I.P.	
1.8.0			W.							
1.9.0		S.W.I.T.C.H.B.	A.							
2.0.0			H.	* -O.O.3						

FORM 274-1152-2
PRINTED IN U.S.A.

INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

IBM

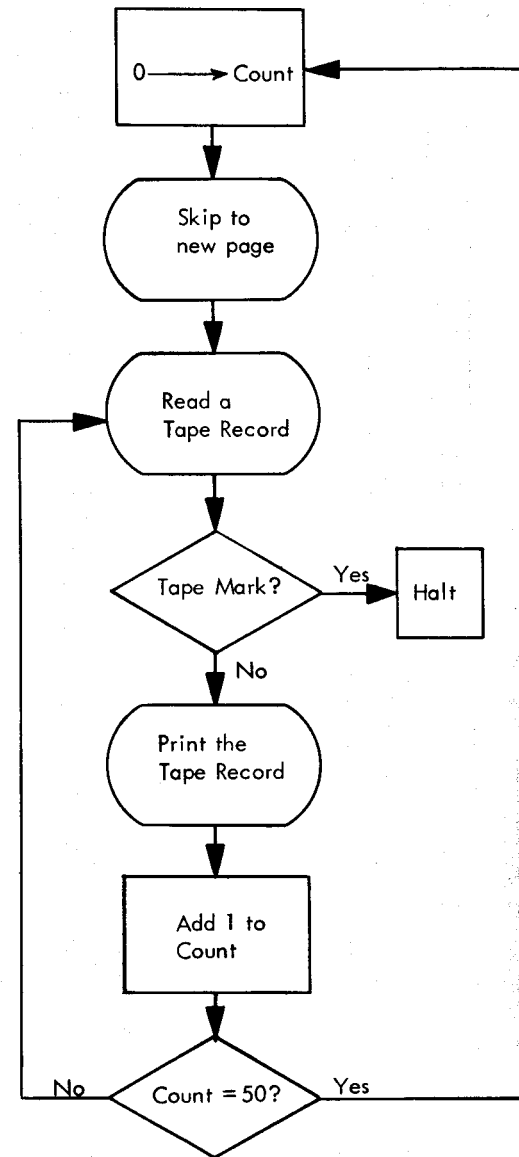
Program _____ Date _____
 Programmed by _____ Identification 78 of 80

LINE	COUNT	LABEL	OPERATION	(A) OPERAND			(B) OPERAND			COMMENTS
				ADDRESS	±	CHAR. ADJ.	ADDRESS	±	CHAR. ADJ.	
3	5 6 7 8	13 14	15 16 17	23	27 28	34	39 40	85		
0.1.0		C.	M.C.W.S.W.O.N.			S.W.I.T.C.H.			S.E.T.S.W.I.T.C.H.O.N.	
0.2.0			B.	D.						
0.3.0	0.6	Z.E.R.O.S	D.C.W.*		O.O.O.O.O.					
0.4.0	0.8	T.O.T.A.L.	D.C.W.*							
0.5.0	0.8	E.D.I.T.	D.C.W.*							
0.6.0	0.5	P.R.A.C.N.O.	D.C.W.*							
0.7.0	0.1	S.W.O.N.	D.C.W.*							
0.8.0			E.N.D.S.T.A.R.T.							

The housekeeping time is so small a fraction of the total job time as to be completely negligible. The most important consideration in estimating this job is whether the processing between cards can be done within the 10 ms of processing time, since the card reader will slow down to 400 cards per minute if not. In this case, however, the processing requires less than 1 ms, so there is no problem. The reading of a five-card group will take $5 \times 75 = 375$ ms. The printing of a summary line for a group requires 100 ms, but to this must be added the waiting time until another starting point in the card reading cycles is read. Since one starting point will have been passed we must wait for another, so that the reading of another card after printing will not begin until 150 ms after the end of reading the card before printing.

The effective time for reading five cards and printing the summary line is therefore $375 + 150 = 525$ ms. With 2,000 groups of five cards, the total time to do the job is thus:

$$2,000 \times 525 \text{ ms} = 1,050,000 \text{ ms} = 1,050 \text{ sec} = 17.5 \text{ min.}$$



INTERNATIONAL BUSINESS MACHINES CORPORATION
IBM 1401 SYMBOLIC PROGRAMMING SYSTEM
CODING SHEET

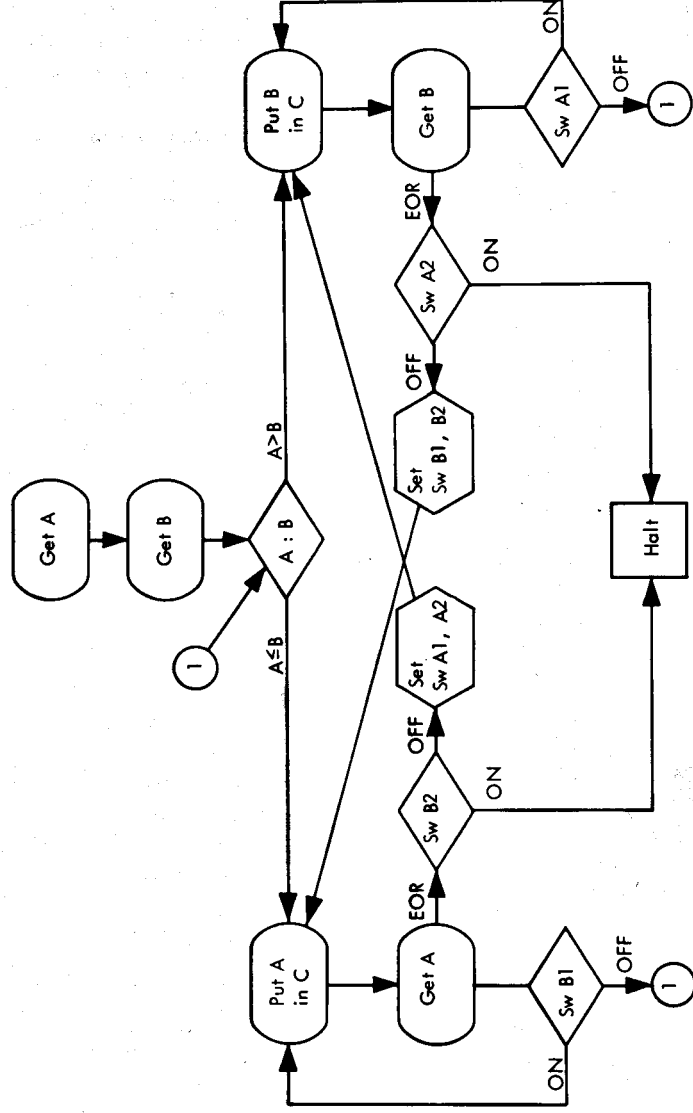
IBM

Program _____ Page No. 2 of 2
 Programmed by _____ Date _____ Identification 76

LINE	COUNT	LABEL	OPERATION	(A) OPERAND		(B) OPERAND		COMMENTS										
				ADDRESS	CHAR. ADJ.	ADDRESS	CHAR. ADJ.											
3	5	6	7	8	13	14	15	16	17	23	27	28	29	34	38	39	40	55
0.1.0		START	M,C,W,Z,E,R,O,S					COUNT										INITIAL COUNTER
0.2.0			C,C															SKIP TO NEW PAGE
0.3.0		READ	M,C,W%U,I					PRINT										READ FROM TAPE
0.4.0			B					HALT										END OF REEL
0.5.0			M,C,W,B,L,A,N,K					PRINT										NO ERASE GROUP
0.6.0			W															PRINT
0.7.0			A					ONE										
0.8.0			C					COUNT										PAGE FULL
0.9.0			B					READ										
1.0.0			B					START										NO
1.1.0			H					HALT										YES
1.2.0	02	Z,E,R,O,S	D,C,W#															
1.3.0	02	COUNT	D,C,W#															
1.4.0		PRINT	D,S					0.2.0.1										
1.5.0	01	B,L,A,N,K	D,C,W#															
1.6.0	01	ONE	D,C,W#															
1.7.0	02	FIFTY	D,C,W#															
1.8.0			E,N,D,S,T,A,R,T															
1.9.0																		
2.0.0																		

In the path coming out of the box "Compare Part Numbers" labeled M>T, insert a test to insure that the transaction code is a zero. If it is, set up the transaction information in master record format, PUT the record in the new master tape, read another card, and return to the comparison. If the code is not zero, write a bad code message. If such an addition record should happen to have the same part number as a master record in the master file, the classification code test will prevent writing it.

Modifying the program should not be difficult and will not be shown.



IBM

Program _____
 Programmed by _____
 Date _____

1401/1410 AUTOCODER CODING SHEET

Identification 76 of 80
 Page No. 12

Line	Label	Operation	Operand
0.1	START	OPEN	A, B, C
0.2		GET	A
0.3		GET	B
0.4	COMP	COMPARE	EMP. NUMBERS
0.5		BL	BRANCH IF B IS LOW
0.6	APUT	PUT	A, C
0.7		GET	A
0.8	B1	NOP	APUT
0.9		B	COMP
1.0	B2	NOP	HALT
1.1		MCW	BRANCH A1
1.2		MCW	BRANCH A2
1.3	BPUT	PUT	B, C
1.4		GET	B
1.5	A1	NOP	BPUT
1.6		B	COMP
1.7	A2	NOP	HALT
1.8		MCW	BRANCH B1
1.9		MCW	BRANCH B2
2.0		B	APUT
2.1	HALT	H	HALT
2.2	BRANCH	DCW	000
2.3		END	START
2.4			
2.5			



Program _____
 Programmed by _____
 Date _____

1401/1410 AUTOCODER CODING SHEET

Identification 76 _____ 80 _____
 Page No. 12 of _____

Line	Label	Operation	25	30	35	40	45	50	55	60	65	70
0.1		MCW	TW0	MULTI+6				SET UP AND				
0.2		M	KEY	MULT				MULTIPLY KEY BY 0.2				
0.3		A	BASE	MULT-1				ADD BASE ADDRESS				
0.4		MCW	MULT-1	DISKAD-1				SET UP SEEK ADDRESS				
0.5		SD	DISKAD-7					SEEK TRACK				
0.6		RD	DISKAD-7					READ DISK				
0.7												
0.8												
0.9												
1.0	TW0	DCW	2									
1.1	BASE	DCW	5000.0									
1.2	MULT	DCW	000000.0									
1.3	DISKAD	DCW	00000000.0									
1.4		DCW										



Program _____
 Programmed by _____
 Date _____

1401/1410 AUTOCODER CODING SHEET

Identification 76 _____ 80 _____
 Page No. 12 of _____

Line	Label	Operation	25	30	35	40	45	50	55	60	65	70
0.1		SW	KEY-2	KEY-5				COMPUTE				
0.2		MCW	KEY-5	WORK				TRACK				
0.3		A	KEY-3	WORK				ADDRESS				
0.4		A	KEY	WORK								
0.5		SW	WORK-2					3 DIGITS ONLY				
0.6		MCW	WORK	DISKAD-2				MOVE TO DISK ADDRESS				
0.7		CW	WORK-2									
0.8		SD	DISKAD-7					SEEK TRACK				
0.9		RD	DISKAD-7					READ TRACK ZERO				
1.0		CW	KEY-2	KEY-5								
1.1		MCW	FIRST	ABC+3				INITIALIZE LOOP ADDRESS				
1.2	ABC	C	0000	KEY				FIND MATCHING KEY				
1.3		BE	FIND									
1.4		A	TEN	ABC+3				MODIFY ADDRESS				
1.5		B	ABC					NO TEST-ASSUME WILL BE FIND				
1.6	FIND	SW	ABC+1					SET UP ADDRESS				
1.7		MCW	ABC+3	DEF+3				OF SECTOR				
1.8		CW	ABC+1					NUMBER				
1.9		SW	DISKAD-1					SET UP				
2.0	DEF	MCW	0000	DISKAD-1				TRACK				
2.1		CW	DISKAD-1					ADDRESS				
2.2		RD	DISKAD-6					READ CORRECT TRACK				
2.3												
2.4												
2.5												

Index

	<i>Section and Page</i>
A-address	3-5
A-address register	3-16
A-register	3-17
Absolute address	4-1, 4-3, 4-11
Access arm	9-2
Accumulator	1-19, 5-10
Actual address—See absolute address	
Add instruction	3-20
Address	3-2, 3-3, 3-13, 4-1, 8-14
Address, disk storage	9-3, 9-15
Address modification	6-1
Address part of instruction	3-5
Algebraic operations	3-22, 5-10
Ampersand	7-2
Assembly	4-7
Assembly listing—See post listing	
Augmented operation code	8-14
Autocoder	4-1, 8-14, 9-5, 11-1
B-address	3-5
B-address register	3-16
B-box—See index register	
B-register	3-15
Backspace tape instruction	8-8
Batch processing	1-19, 9-1
Binary digit	2-22
Bit	2-22, 8-1
Blank (character)	3-6, 3-9, 7-2
Block count	8-16
Block diagram	1-4, 1-12, 1-16, 5-12
Block, magnetic tape	8-1, 8-15
Blocking, of tape records	8-15
Blocking factor	8-16, 8-31
Body of control word	7-3

	<i>Section and Page</i>
Branch If Character Equal instruction	8-27
Branch If Indicator On instruction	3-21, 5-1, 5-2, 5-9, 7-9, 8-9, 9-10
Branch If Word Mark and/or Zone instruction	5-8, 5-10
Branch instruction	4-2, 4-5, 4-16, 5-1
Brush	2-15
Buffering	7-16
Calculation	1-2
Card column	2-1
Card loader	7-22
Card punch	1-13
Card punching	2-7, 7-13, 11-19
Card reading	2-6, 3-5, 7-11, 11-6, 11-19
Card to Disk routine	9-18
Card-to-tape	2-10
Card verifier	1-13
Carriage control	5-2, 7-6
Central processing unit	2-5
Chain Additions Program	9-19
Chain Loading Program	9-19
Chain Maintenance Program	9-19
Chain printing mechanism	2-9
Chaining, disk storage	9-15
Chaining, of instruction addresses	3-18
Channel, carriage tape	7-7
Character	2-2, 2-22, 3-2
Character adjustment	4-5
Character coding	2-22, 8-1, 8-2
Character density, magnetic tape	8-4

	<i>Section and Page</i>
Character rate, magnetic tape	8-4
Checking—See error checking	
Checkout, program	1-5, 10-13
Clear Disk Storage routine	9-18
Clear Storage instruction	3-9, 3-10
Clear Storage program	7-22
Clear Word Mark instruction	3-7, 3-8
CLOSE macro-instruction	8-15, 8-19, 11-19
Closed subroutine	7-21
COBOL	11-6, 11-12
Coding	1-4, 10-18, 11-6
Collating	1-4, 5-17
Collator	1-7, 2-17
Color stripe	2-3
Column, card	2-1
Comma	7-2
Comments	4-22, 11-16
Compare instruction	5-9, 5-10
Comparison	5-2, 5-10
Compatibility	2-11
Compiler	8-15
Computation section of loop	6-7
Conditional branch	5-1, 11-6, 11-14
Console, 1401 Processing Unit	2-5, 7-22
Control card	7-23
Control Carriage instruction	7-7, 7-8, 7-9, 7-13
Control field	1-3, 1-19
Control level	1-19
Control panel	2-3, 2-18
Control section	2-14, 3-15
Control total	1-15, 2-10, 8-17, 10-4
Control Unit instruction	8-8
Conversion, file	1-5, 10-4
Core	2-22

	<i>Section and Page</i>
Corner cut card	2-3
Count field	4-5
Credit amount	7-2
d-character	3-5, 3-16, 5-1, 8-14
DA macro-instruction	8-20, 8-31
Data division, COBOL	11-12
Data processing	1-2
DC pseudo instruction	4-11
DCW pseudo instruction	4-5
Deblocking	8-16
Decimal point	7-2
Decision	1-4, 3-5, 5-2, 11-2
Decision table	11-1
Define Area macro-instruction	8-20, 8-31
Define Constant	4-11
Define Constant with a Word Mark	4-5
Define IOCS macro-instruction	8-15
Define Symbol	4-7
Define Tape File macro-instruction	8-15, 8-18, 8-27
Density, magnetic tape	8-4
Desk checking	10-13
Detail file	1-7
DIOCS macro-instruction	8-15
Disk file utility routines	9-18
Disk storage	9-2
Disk storage addressing	9-3, 9-15
Disk storage instructions	9-5
Disk to Card routine	9-18
Disk to Tape routine	9-18
Documentation	10-17
Dollar sign	7-2
DS pseudo instruction	4-7
DTF macro-instruction	8-15, 8-18, 8-27
Dual level sensing	2-12
E-phase	3-17, 6-1

	<i>Section and Page</i>
Early card read	7-11
Editing	1-2, 2-10, 7-1
Effective address	6-20
End-of-reel spot	5-2, 8-5
END pseudo instruction	4-7, 4-13, 7-22
Environment division, COBOL	11-12
EQU instruction	8-23
Equal compare	5-3, 5-9
Error checking	2-6, 2-9, 2-20, 5-2, 10-4
Error routine	8-19
Execute phase	3-17, 6-1
Execution of instructions	3-17, 6-1
Expanded print edit	7-4
Extension, of an amount	1-12, 1-21, 2-20
External storage	3-1
Field	2-3, 3-1
File	1-3
File conversion	1-5, 10-4
File, magnetic tape	8-5
File protection ring	8-5
Fixed length record	8-16
Fixed word length	3-2
Flow chart	1-4, 1-12, 11-2
Form control	7-6
Form design	7-1, 10-13
Format	7-1
FORTRAN	11-4
Full track read	9-6
Function, FORTRAN	11-6
GET macro-instruction	8-15, 8-19, 11-19
Grocery wholesaler example	9-20
Group mark	8-7, 8-11, 8-13, 8-20, 9-7
Halt instruction	5-7, 5-12
Hardware	2-1
Hash total	8-17
Head, magnetic tape	8-3
Header label	8-17
Heading line	7-10
High compare	5-3, 5-9
High-Low-Equal compare device	5-9
High-order position	3-13, 4-8
Hole count	2-7
Home record, in disk storage	9-16
Horizontal check character	8-2
I-address register	3-16
I-phase	3-17, 6-1
IBM 83 Card Sorter	2-15
IBM 85 Collator	2-17
IBM 729 Tape Unit	2-11, 8-3, 8-4
IBM 1301 Disk Storage Unit	9-2, 9-4
IBM 1401 Data Processing System	1-1, 2-5
IBM 1402 Card Read Punch	2-6, 7-11, 7-12, 7-13
IBM 1403 Printer	2-8, 7-6, 7-13
IBM 1405 Disk Storage Unit	2-12, 9-2, 9-3
IBM 1407 Console Inquiry Station	2-13, 2-14
IBM 7330 Tape Unit	2-11, 8-3, 8-4
IBM Charting and Diagramming Templates	1-13
IBM punched card	2-1
Index register	6-19, 7-21
Indexed disk storage	9-17
Indexing accumulator—See index register	
Initialization Section of loop	6-7, 6-10
Input/Output Control System	8-11, 8-14, 8-18, 11-19

	<i>Section and Page</i>
Instruction	1-4, 3-1, 3-5, 3-13, 5-7, 6-1
Instruction phase	3-17, 6-1
Internal storage	3-1
Interpreter	2-3
Interrecord gap	8-1, 8-4
Inventory control example	8-23, 9-10, 10-1, 11-2, 11-19
IOCS—See Input/Output Control System	
Key—See control field	
Keypunch—See card punch	
Label tape	8-17
Label table	4-12
Last card switch	5-2, 5-3, 5-12
Line number	4-8
Linkage, subroutine	7-21, 11-18
Listing	1-2
Literals	8-14, 11-16, 11-18
Load button	7-22
Load Characters to A Word Mark	4-15
Load point, magnetic tape	8-5
Location	3-2
Location counter	4-12
Loop	6-1, 6-7
Low compare	5-3, 5-9
Low-order position	3-2, 4-8
Machine-oriented language	11-6
Macro-instruction	8-6, 8-15
Magnetic core	2-22, 3-2
Magnetic tape	2-10, 2-14, 8-1
Master file	1-3, 2-10, 8-23
Master file creation	10-4, 10-16
Memory—See storage	
Memory dump routine	7-23, 10-14
Merging	1-4
Minus sign	2-24, 7-2
Mnemonic operation code	4-3, 8-14
Modification section of loop	6-8
Move Characters and Edit instruction	4-19, 4-20, 7-2
Move Characters and Suppress Zeros instruction	5-4, 7-2
Move Characters to A or B Word Mark Instruction	3-5, 3-7
Multiply instruction	4-15, 4-18
Negative quantity	7-2
No-Operation instruction	6-6
Normal punch pocket	2-8
Normal read pocket	2-7
Numerical bits	2-22
Numerical punch	2-1
Object program	4-1, 4-7, 4-11, 8-15, 11-4
OP-register	3-16
OPEN macro-instruction	8-15, 8-18, 11-19
Open subroutine	7-21
Operating instructions	10-18
Operation code	3-5
ORG pseudo instruction	4-11
Origin	4-11
Overflow	5-3
Overflow record, in disk storage	9-16
Page heading	7-10
Page number	4-8
Parallel operation	10-16
Parity bit	2-22
Parity checking	2-12, 2-23, 8-2
Partial chaining	3-19
Parts explosion and summary example	5-15

	<i>Section and Page</i>
Pass	2-16, 4-12
Payroll example	4-14
Phase, of instruction	
execution	3-17
Photo-sensing markers,	
magnetic tape	8-5
Picking sequence,	
warehouse	9-20, 9-22
Pilot operation	10-15
Plus sign	2-24
Post-listing	4-7, 10-14
Print area	3-9
Print storage	5-2, 7-17
Printer carriage control	7-6
Printing	2-8, 7-13, 11-19
Procedure design	1-1
Procedure division,	
COBOL	11-12
Procedure-oriented	
language	11-6
Procedures manual	10-18
Processing time	7-11
Processing unit	2-5, 2-14, 3-16
Processor	4-1
Program	1-4, 3-1
Program checkout	10-13
Program identification	4-8
Program loading	7-22
Program	
switch	6-5, 8-24, 8-25
Programming	1-4, 11-1
Pseudo instruction	4-11
Punch a Card	
instruction	3-9, 7-13
Punch area	3-9
Punch feed read	2-8
Punch release	7-13
Punch start time	7-13
PUT macro-instruction	8-15, 8-19, 11-19
RAMAC	9-2
Random access file	
processing	1-20

	<i>Section and Page</i>
Random access storage	1-7, 1-19, 2-12, 2-14, 9-1
Read a Card	
instruction	3-8, 7-11
Read and Punch	
instruction	7-15
Read area	3-8
Read Disk instruction	9-6, 9-7
Read release	7-11
Read start time	7-11
Read Tape	
instruction	8-6, 8-7, 8-8
Record	1-3
Record, magnetic	
tape	8-1, 8-15
Record count	8-16
Reel, magnetic tape	8-1, 8-5
Reflective spot, magnetic	
tape	8-5
Register	3-15, 3-16
Relative sensitivity level	8-2
Report	7-1
Report file	1-3
Report Program	
Generator	11-7
Restart procedure	8-17
Rewind, magnetic tape	8-4
Rewind Tape instruction	8-10
Rewind Tape and Unload	
instruction	8-10, 8-11
Rounding	4-4
Routine—See program	
Run manual	10-17
Sales statistics	
example	1-6, 1-21
Sector, disk storage	9-3
Seek Disk instruction	9-5, 9-6
Select Stacker instruction	5-20
Sense switch	5-2
Sequential access file	1-7
Sequential file processing	1-6, 1-19, 8-23, 9-1, 10-7
Set Word Mark instruction	3-7

	<i>Section and Page</i>
Sign	2-24, 3-21, 5-10
Skip and Blank Tape	
instruction	8-9
Skipping, of form	7-6, 7-13
Sorter	1-7, 2-15
Sorting	1-3, 1-7, 1-19, 2-10, 2-16, 5-17, 8-24, 9-1, 10-6
Source data	1-2, 1-13
Source program	4-1, 4-7, 4-11, 8-15, 11-2, 11-4, 11-12
Spacing, of form	7-6, 7-13
Speed—See timing	
SPS—See symbolic	
programming system	
Stacker selection	2-7, 2-20, 5-19
Stacking	1-12, 2-7
Start and stop time, magnetic	
tape	8-4
Start button	5-7
Status portion of edit word	7-3
Storage	3-1
Storage of instructions	3-13
Storage print	
routine	7-23, 10-14
Store B-Address Register	
instruction	7-21
Stored program computer	6-2
Subroutine	7-20, 11-18
Subtract instruction	3-22
Summarization	1-2
Summary line	7-10
Switch, program	6-5, 8-24, 8-25
Symbolic address	4-1, 4-2
Symbolic instruction	4-11
Symbolic Programming	
System	4-1, 7-22, 11-1, 11-6
System analysis and	
design	1-1, 2-20, 11-1
Tape, carriage	7-6
Tape blocking	8-15
Tape-controlled	
carriage	5-2, 7-6

	<i>Section and Page</i>
Tape head	8-3
Tape label	8-17
Tape mark	8-5, 8-8
Tape-to-card	2-10
Tape to Disk routine	9-18
Tape unit	8-3
Template, charting and	
diagramming	1-13
Test data	10-14
Testing, program	1-5, 10-14
Testing section of loop	6-8
Three-character	
address	3-3, 4-3, 4-13
Timing, input and	
output	2-6, 7-10
Timing, instruction	7-20
Timing, magnetic tape	8-4
Timing, program	7-18
Tracing	10-15
Track, disk storage	9-3
Trailer label	8-16
Transaction file	1-3, 1-6
Transition card	4-13, 7-22
Two-gap head	2-12, 8-3
Unconditional branch	4-5, 5-1, 11-6, 11-16
Unequal compare	5-2, 5-9
Unit record equipment	1-19
Unmatched detail	1-8, 2-20
Unmatched master	1-7, 5-18
Updating, of files	1-3
Utility	
programs	7-20, 7-22, 9-18
Variable instruction length	3-5
Variable length record	8-16
Variable word length	3-2
Verifier	1-13
Wholesale grocery	
example	9-20
Word	3-1

	<i>Section and Page</i>
Word mark	2-22, 2-23, 3-2, 3-5, 3-11, 3-13, 5-2, 5-12, 8-7, 9-10, 9-24
Work-flow chart	1-12
Working storage	8-16
Write a Line instruction	3-9, 7-13, 7-17
Write and Branch instruction	4-5
Write and Punch instruction	7-15
Write and Read instruction	7-15
Write Disk instruction	9-6, 9-8

	<i>Section and Page</i>
Write Disk Check instruction	9-9
Write, Read and Punch instruction	7-15
Write Tape instruction	8-6, 8-7
Write Tape Mark instruction	8-8, 8-9
Zero and Add instruction	6-10
Zero suppression	4-19, 5-3, 7-2
Zero zone	2-1
Zone bits	2-22, 3-21
Zone punch	2-1