

TECHNICAL INFORMATION EXCHANGE

20 MAR 1969

IBM®

November 13, 1968

RPG PROGRAM OPTIMIZATION TECHNIQUES

W. E. Priddy
IBM Corporation
6900 Fannin
Houston, Texas 77025

This paper consists of programming tips and charts to aid RPG programmers in improving core utilization and execution speed. A System 360 Model 20 with 16K, MFCM, Tapes, and Disks was used for testing. Three compilers were used -- Card (V2-L4); Tape (V2-L2); and Disk (V2-L0). The contents of this paper are subject to change with different compiler releases. Numerous changes were incorporated in the Disk Compiler which resulted in an approximate 20% improvement in core utilization.

*is this valid
for DS?*

For IBM Internal Use Only

360 Model 20 RPG

Suggested Methods of Improving Throughput and Core Utilization

1. Use packed numeric fields in records as much as possible. This will reduce core and improve throughput since RPG will not have to generate or execute instructions to pack/unpack the input/output fields. This will also improve the read-write time on tape and disk. It can also reduce the time for reading and punching cards. You can save core too, by reducing the size of the input/output areas.
2. An ADD, SUB, Z-ADD, Z-SUB, MULT, DIV, or COMP between fields or constants with unequal decimal positions will take 6 to 40 bytes more than having equal decimal positions. It can also take 7 times longer to execute. (see charts)
3. A COMP between alpha fields will execute 7 times faster than numeric fields. There is also an 18 byte savings under Card or Tape. Sometimes you may have to define the input field as both numeric and alpha.
4. If several calculations are conditioned by the same indicator(s), group them together. Tape and Disk will generate only one test. Sometimes Card will too. (see chart)
5. Multiplying by the reciprocal instead of dividing can save core.
6. See the last 3 pages of charts for examples of how to save core and execution time when you have several Compares to perform.
7. With Card and Tape you will save 24 bytes by not having 3 field names in an ADD or SUB.

Wrong:	A	add	B	C	-	36	bytes
Right:	A	add	C	C	-	6	bytes
	B	add	C	C	-	6	bytes
8. Instead of Z-ADD 0 Field (Field has 2 decimals)
use Z-ADD 0.00 Field (Saves 24 to 28 bytes)
9. If you have calculations at Total time, you can save core by coding:


```
L0 NL1      Goto End
L1
L2 15
L3 01
L0      End Tag
```

10. Reduce the number of Indicators used on Output as much as possible.

Wrong:	D	01	02	MR
	D	01	02	MR
	D	01	02	MR
Right:	C	01	02	MR Seton 03
	O	D		03
		D		03
		D		03

This will save 14 bytes on Card and 20 on Tape or Disk. You will also save the testing of up to 6 indicators on every cycle.

11. You can save core by reducing the number of edit words. Make your fields have the same field size.
12. Listing your Output fields in descending sequence by ending position can save time.
13. You can save 6 bytes per edit word or zero suppress field if you leave one space between the first print position of the field and the last position of the previous field (if the unpacked field length is ODD; leave 2 spaces if the unpacked field length is EVEN).
14. A "B" in column 22 of the "H" Control Card will create an additional input area for the 2501 and can significantly increase throughput. (Up to double)
15. When stating the size of the Printer in the "H" Control Card and File Description, specify the last print position you use. This can save core and increase throughput. The same holds true for Card Readers and Punches.
16. You can save up to 2K of Core on Tape and Disk by overlaying the OPEN routine. A "1", "D", or "B" in column 32 of the "H" Control Card will allow Overlay. (See page 14 of Disk Performance Estimates C33-6003-0)

1 - Tape
D - Disk
B - Both

17. On Input Specifications specify the most frequent type of record in a file first, then the next most frequent. This will save execution time since RPG tests records in sequence.
18. On Calculation and Output Specifications, conditioned by more than one indicator, code the most likely indicator first. This will save execution time also.
19. If records in a file have basically the same format, use an OR condition and Field Record Relation indicators on Input Specifications. The same principal applies on Output Specifications. This will save core.
20. If you only want to process certain types of records in a file, you can save execution time this way:

Wrong: ICARDIN AA 01 80C1
or 02

Field A
Field B

Right: ICARDIN AA 01 80C1

Field A
Field B

AB 02

This will save the time required to MOVE the unwanted records into Field A and B. Also it will not test MR and Levels.

21. On Input Spec. use "C" instead of "Z" or "D" for Record Identification. This will save core and time.
22. If matching record control fields are in the same positions in all records, it will save core. Use the same name in all records if possible.
23. Instead of splitting control fields (2 fields defined as L1) use L1 and L2. Saves core.
24. On Output use Space/Skip After instead of Before. Saves time.
25. If you are using Tables on a Disk Model 20, be sure to specify a "B" in column 34 of the "H" Control Card for Binary search. Saves time.

26. Card Model 20's can benefit from #25 by compiling with the Disk Compiler and executing their object deck using the Card IPL, Job Control, and Monitor on their Card 20. They would also have the ability to load and update the table during execution.

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System				Comments		
									Card V2/L4	Tape V2/L2	Disk V2/L0				
									Instruct. Bytes	Instruct. Bytes	Instruct. Bytes				
	A	ADD	B	B	=	=			AP	6	Same		Same	Two Field Names	
	B	ADD	A	B	=	=			AP	6	Same		Same		
	L	ADD	A	L	≠	=			AP	6	Same		Same	Unequal Field Lengths cause no change in core if the Result field is the larger.	
	A	ADD	L	L	≠	=			AP	6	Same		Same		
	A	ADD	K	K	≠	≠			BAS	4	Same	46	BAS		4
									MVC	6	As		MVO		6
									ZAP	6	Card		MVO		6
									ZAP	6			MVO		6
									MVN	6			AP		6
									NI	4			BCR		2
									AP	6					30
									ZAP	6					
									BCR	2					
										46					
	A	ADD	C	C	=	≠			BAS	4	Same		BAS	4	
									MVC	6	As	46	MVO	6	
									ZAP	6	Card		MVO	6	
									ZAP	6			MVO	6	
									MVN	6			AP	6	
									NI	4			BCR	2	
									AP	6				30	
									ZAP	6					
									BCR	2					
										46					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System				Comments
									Card V2/L4	Tape V2/L2	Disk V2/L0		
									Instruct. Bytes	Instruct. Bytes	Instruct. Bytes		
** ** *									The following pages will show the generated instructions for RPG Source Program Calculation Instructions. The purpose is to determine the coding technique that will use the least amount of core. It has been found that certain instructions execute much faster than others. The timings are on another chart. You will find that Calculations between fields with unequal decimal positions and unequal field lengths will quite often make a difference in core utilization and execution time.				
** ** *									Field Definitions:				
				A	5	0							
				B	5	0							
				C	5	2							
				D	5	2							
				E	5	0							
				F	5	2							
				G	5	Alpha							
				H	5	Alpha							
				I	10	Alpha							
				J	10	0							
				K	10	2							
				L	10	0							

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System			Comments		
									Card V2/L4	Tape V2/L2	Disk V2/L0			
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes
	C	SUB	A	C	=	#			BAS	4	Same		BAS	4
									MVC	6	As		MVO	6
									ZAP	6	Card		MVO	6
									ZAP	6			MVO	6
									MVN	6			SP	6
									NI	4			BCR	2
									SP	6				30
									ZAP	6				
									BCR	2				
										46				
	J	SUB	A	J	#	=			SP	6	Same		Same	
	A	SUB	B	E	=	=			BAS	4	Same		ZAP	6
									MVC	6	As		SP	6
									ZAP	6	Card			12
									ZAP	6				
									SP	6				
									ZAP	6				
									BCR	2				
										36				
	C	SUB	A	D	=	#			BAS	4	Same		BAS	4
									MVC	6	As		MVO	6
									ZAP	6	Card		MVO	6
									ZAP	6			MVO	6
									MVN	6			ZAP	6
									NI	4			SP	6
									SP	6			ZAP	6
									ZAP	6			BCR	2
									BCR	2				42
										46				

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System			Comments		
									Card V2/L4	Tape V2/L2	Disk V2/L0			
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes
	A	ADD	B	E	=	=			BAS	4	Same	36	ZAP	6
									MVC	6	As		AP	6
									ZAP	6	Card			12
									ZAP	6				
									AP	6				
									ZAP	6				
									BCR	2				
										36				
	A	ADD	C	D	=	#			BAS	4	Same		BAS	4
									MVC	6	As		MVO	6
									ZAP	6	Card		MVO	6
									ZAP	6			MVO	6
									MVN	6			AP	6
									NI	4			ZAP	6
									AP	6			BCR	2
									ZAP	6				36
									BCR	2				
										46				
	A	ADD	J	L	#	=			BAS	4	Same		ZAP	6
									MVC	6	As		ADD	6
									ZAP	6	Card			12
									ZAP	6				
									AP	6				
									ZAP	6				
									BCR	2				
										36				
	A	SUB	B	A	=	=			SP	6	Same		Same	

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System			Comments		
									Card V2/L4 Instruct. Bytes	Tape V2/L2 Instruct. Bytes	Disk V2/L0 Instruct. Bytes			
01	A	SUB	B	A	=	=			CLI BC SP	4 4 6	Same As Card		Same	One Conditioning Indicator
										14				
02 03	A	SUB	B	A	=	=			CLI BC CLI BC SP	4 4 4 4 6	Same As Card		Same	2 Conditioning Indicator
										22				
04 05 06	A	SUB	B	A	=	=			CLI BC CLI BC CLI BC SP	4 4 4 4 4 4 6	Same As Card		Same	3 Conditioning Indicator
										30				
		seton						01	BAS MVI BCR	4 4 2	MVI	4	MVI	4
										10				
		seton						01 02	BAS MVI MVI BCR	4 4 4 2	MVI MVI	4 4	Same As Tape	
										14				

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System			Comments		
									Card V2/L4 Instruct. Bytes	Tape V2/L2 Instruct. Bytes	Disk V2/L0 Instruct. Bytes			
	A	SUB	B	A	=	=		01	SP MVI BC MVI	6 4 4 4	Same		Same	One Resulting Indicator
										18				
	A	SUB	B	A	=	=		02 03	SP MVI BC MVI MVI BC MVI	6 4 4 4 4 4 4	Same As Card		Same	2 Resulting Indicators
										30				
	A	SUB	B	A	=	=		04 05 06	SP MVI BC MVI MVI BC MVI BC MVI	6 4 4 4 4 4 4 4 4	Same As Card		Same	3 Resulting Indicators
										42				

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper- ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System			Comments		
									Card V2/L4 Instruct. Bytes	Tape V2/L2 Instruct. Bytes	Disk V2/L0 Instruct. Bytes			
	A	mult	B	J	#	=			BAS 4 MVC 6 ZAP 6 MP 6 ZAP 6 BCR 2	Same As Card	ZAP 6 MP 6 ZAP 6	6 6 6 18		
	A	mult	C	E	#	#			BAS 4 MVC 6 ZAP 6 MP 6 MVN 6 ZAP 6 BCR 2	Same As Card	ZAP 6 MP 6 MVN 6 ZAP 6	6 6 6 6 24		
	A	mult	C	E	#	#	H		BAS 4 MVC 6 ZAP 6 MP 6 ZAP 6 NI 4 AP 6 MVN 6 ZAP 6 BCR 2	Same As Card	BAS 4 ZAP 6 MP 6 ZAP 6 NI 4 AP 6 MVN 6 ZAP 6 BCR 2	4 6 6 6 4 6 6 6 2 46	Half Adjust. takes 16 bytes	
									30					
									36					
									52					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper- ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System			Comments		
									Card V2/L4 Instruct. Bytes	Tape V2/L2 Instruct. Bytes	Disk V2/L0 Instruct. Bytes			
		seton						01 02 03	BAS 4 MVI 4 MVI 4 MVI 4 BCR 2	MVI 4 MVI 4 MVI 4	4 4 4 12	Same As Tape		
		seton						01 02 03	BAS 4 MVI 4 MVI 4 MVI 4 BCR 2	MVI 4 MVI 4 MVI 4	4 4 4 12	Same As Tape		
		setof						01 02 03	BAS 4 MVI 4 MVI 4 MVI 4 BCR 2	MVI 4 MVI 4 MVI 4	4 4 4 12	Same As Tape		
	A	mult	B	E	=	=			BAS 4 MVC 6 ZAP 6 MP 6 ZAP 6 BCR 2	Same As Card	ZAP 6 MP 6 ZAP 6	6 6 6 18		
									30					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
	J	DIV	A	L	#	=			BAS	4	Same		ZAP	6	
									MVC	6	As		DP	6	
									ZAP	6	Card		ZAP	6	
									ZAP	6				18	
									DP	6					
									ZAP	6					
									BCR	2					
										36					
		Z-Add	0	A	=	=			ZAP	6	Same		Same		
		Z-Add	0	C	=	#			BAS	4	Same		BAS	4	Unequal Decimal Positions
									MVC	6	As		MVO	6	
									ZAP	6	Card		MVO	6	
									MVN	6			MVO	6	
									NI	4			ZAP	6	
									ZAP	6			BCR	2	
									BCR	2				30	
										34					
		Z-Add	0.00	C	=	=			ZAP	6	Same		Same		Equal Decimal Positions
		Z-Add	000.00	C	=	=			ZAP	6	Same		Same		
		Z-Add	A	B	=	=			ZAP	6	Same		Same		

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
	A	Div	B	E	=	=			BAS	4	Same		ZAP	6	
									MVC	6	As		DP	6	
									ZAP	6	Card		ZAP	6	
									ZAP	6				18	
									DP	6					
									ZAP	6					
									BCR	2					
										36					
	A	Div	C	D	=	#			BAS	4	Same		BAS	4	
									MVC	6	As		MVO	6	
									ZAP	6	Card		MVO	6	
									MVN	6			MVO	6	
									NI	4			DP	6	
									ZAP	6			ZAP	6	
									DP	6			BCR	2	
									ZAP	6				36	
									BCR	2					
										46					
	A	Div	C	E	#	#	H		BAS	4	Same		BAS	4	
									MVC	6	As		MVO	6	
									ZAP	6	Card		MVO	6	
									MVN	6			DP	6	
									NI	4			AP	6	
									MVO	6			MVO	6	
									ZAP	6			ZAP	6	
									DP	6			BCR	2	
									AP	6				42	
									MVO	6					
									ZAP	6					
									BCR	2					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper- ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
		Z-Sub	A	C	=	#			BAS	4	Same		BAS	4	
								MVC	6	As		MVO	6		
								ZAP	6	Card		MVO	6		
								SP	6			MVO	6		
								MVN	6			ZAP	6		
								NI	4			SP	6		
								ZAP	6			BCR	2		
								BCR	2				2		
									40				36		
		Z-Sub	A	K	#	#			BAS	4	Same		BAS	4	
								MVC	6	As		MVO	6		
								ZAP	6	Card		MVO	6		
								SP	6			MVO	6		
								MVN	6			ZAP	6		
								NI	4			SP	6		
								ZAP	6			BCR	2		
								BCR	2				2		
									40				36		
		Move	A	B	=	=			BAS	4	MVC	6	MVC	6	
								MVC	6						
								BCR	2						
									12						
		Move	A	C	=	#			BAS	4	MVC	6	MVC	6	
								MVC	6						
								BCR	2						
									12						

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper- ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
		Z-Add	A	C	=	#			BAS	4	Same		BAS	4	
								MVC	6	As		MVO	6		
								ZAP	6	Card		MVO	6		
								MVN	6			MVO	6		
								NI	4			ZAP	6		
								ZAP	6			BCR	2		
								BCR	2				2		
									34				30		
		Z-Add	A	K	#	#			BAS	4	Same		BAS	4	
								MVC	6	As		MVO	6		
								ZAP	6	Card		MVO	6		
								MVN	6			MVO	6		
								NI	4			ZAP	6		
								ZAP	6			BCR	2		
								BCR	2				2		
									34				30		
		Z-Sub	A	B	=	=			ZAP	6	Same		ZAP	6	
								ZAP	6	As		SP	6		
								SP	6	Card			6		
									18				12		

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
	A	Comp	B	=	=			01 02	BAS MVC ZAP ZAP SP BCR MVI BC MVI MVI BC MVI	4 6 6 6 6 2 4 4 4 4 4 4	Same As Card		CP MVI BC MVI BC MVI	30	Numeric COMP -2 Indicators
	A	Comp	B	=	=			01 02 03	BAS MVC ZAP ZAP SP BCR MVI BC MVI MVI BC MVI BC MVI BC MVI	4 6 6 6 6 2 4 4 4 4 4 4 4 4 4	Same As Card		CP MVI BC MVI MVI BC MVI BC MVI	42	Numeric COMP-3 Indicator

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments	
									Card V2/L4		Tape V2/L2		Disk V2/L0			
		Move	A	J	≠	=			BAS MVC BCR	4 6 2		MVC	6	MVC	6	
		Move	G	H	=	Alpha			BAS MVC BCR	4 6 2		MVC	6	MVC	6	
		Move	G	I	≠	Alpha			BAS MVC BCR	4 6 2		MVC	6	MVC	6	
		Move	A	G	=	Num and Alpha			BAS UNPK BCR	4 6 2		UNPK	6	UNPK	6	
	A	Comp	B	=	=			01	BAS MVC ZAP ZAP SP BCR MVI BC MVI	4 6 6 6 6 2 4 4 4	Same As Card		CP MVI BC MVI	18	Numeric COMP - 1 Indicator	

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
		G	Comp	H		=	Alpha	02 03	BAS	4	CLC	6	Same	Each Resulting Indicator takes 12 bytes.	
									CLC	6	MVI	4	As		
									BCR	2	BC	4	Tape		
									MVI	4	MVI	4			
									BC	4	MVI	4			
									MVI	4	BC	4			
									MVI	4	MVI	4			
									BC	4		30			
									MVI	4					
										36					
		G	Comp	H		=	Alpha	04 05 06	BAS	4	CLC	6	Same		
									CLC	6	MVI	4	As		
									BCR	2	BC	4	Tape		
									MVI	4	MVI	4			
									BC	4	MVI	4			
									MVI	4	BC	4			
									MVI	4	MVI	4			
									BC	4	MVI	4			
									MVI	4	BC	4			
									MVI	4	MVI	4			
									BC	4		42			
									MVI	4					
										48					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
		A	Comp	C		=	≠	01	BAS	4	Same		BAS	4	Notice increase in core when decimal positions are unequal.
									MVC	6	As		MVO	6	
									ZAP	6	Card		MVO	6	
									ZAP	6			MVO	6	
									MVN	6			CP	6	
									NI	4			BCR	2	
									SP	6			MVI	4	
									BCR	2			BC	4	
									MVI	4			MVI	4	
									BC	4				42	
									MVI	4					
										52					
		A	Comp	J		≠	=	01	BAS	4	Same		ZAP	6	No increase in core for ≠ Field Lengths with Card or Tape. Takes 6 bytes extra on Disk.
									MVC	6	As		CP	6	
									ZAP	6	Card		MVI	4	
									ZAP	6			BC	4	
									SP	6			MVI	4	
									BCR	2				24	
									MVI	4					
									BC	4					
									MVI	4					
										42					
		G	Comp	H		=	Alpha	01	BAS	4	CLC	6	Same	Alpha COMP-18 bytes less than Numeric COMP on Card. 24 less for Tape and none for Disk. Alpha COMP will execute faster than Numeric COMP.	
									CLC	6	MVI	4	As		
									BCR	2	BC	4	Tape		
									MVI	4	MVI	4			
									BC	4		18			
									MVI	4					
										24					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
		Next	Goto Tag	Next					BAS	4	BAS	4	BC	4	Disk uses "BC". Tape and Card use "BAS"
								---	-	---	-	--	-		
01		A	Add	B	B				CLI(01)	4	Same	14	Same	14	Test generated code for conditioning indicators.
								BC	4						
									AP	6					
01		A	Add	B	B				CLI(01)	4	AP	6	AP	6	
									BC	4					
									AP	6					
01	02	A	Add	B	B				CLI(01)	4	CLI(02)	4	CLI(01)	4	
									BC	4	BC	4	BC	4	
									CLI(02)	4	AP	6	CLI(02)	4	
									BC	4			BC	4	
									AP	6			AP	6	
01	03	A	Add	B	B				CLI(01)	4	CLI(03)	4	Same	14	
									BC	4	BC	4			
									CLI(03)	4	AP	6			
									BC	4					
									AP	6					
01	03	A	Add	B	B				AP	6	AP	6	AP	6	
01	04	A	Add	B	B				CLI(01)	4	CLI(04)	4	Same	14	
									BC	4	BC	4			
									CLI(04)	4	AP	6			
									BC	4					
									AP	6					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition. Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
		G	Comp	I		≠	Alpha	01	BAS	4	MVI	4	BAS	4	Unequal Alpha field length adds 16 bytes on Card and Tape. Adds 18 on Disk.
									MVI	4	MVC	6	MVC	6	
									MVC	6	MVC	6	MVC	6	
									MVC	6	CLC	6	CLC	6	
									CLC	6	MVI	4	BCR	2	
									BCR	2	BC	4	MVI	4	
									MVI	4	MVI	4	BC	4	
									BC	4		34	MVI	4	
									MVI	4				36	
										40					
01		A	Add	B	B	=	=		CLI(01)	4	Same		Same		Test coding for Conditioning Indicators.
									BC	4					
									AP	6					
01		A	Add	B	B				AP	6	Same		Same		
01	02	A	Add	B	B				CLI(01)	4	CLI(02)	4	Same		
									BC	4	BC	4	As		
									CLI(02)	4	AP	6	Tape		
									BC	4					
									AP	6					
01	02	A	Add	B	B				AP	6	Same		Same		
										48		40		40	

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4	Tape V2/L2	Disk V2/L0				
								Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes		
METHOD B															
	G	Comp	'00000'			=	Alpha	99	BAS	4	CLC	6	Same	18	Alpha COMP-save 18-24 bytes per compare on Card and Tape. These instructions will accomplish the same as above. Define the field on input as Numeric (A) and Alpha(G).
									CLC	6	MVI	4	As		
									BCR	2	BC	4	Tape		
									MVI	4	MVI	4			
									BC	4					
									MVI	4					
	G	Comp	'00001'					01		24		18		18	
	G	Comp	'00002'					02		24		18		18	
	G	Comp	'00003'					03		24		18		18	
	G	Comp	'00004'					04		24		18		18	
	G	Comp	'00005'					05		24		18		18	
	G	Comp	'00006'					06		24		18		18	
	G	Comp	'00007'					07		24		18		18	
										192		144		144	
METHOD C															
		setof						03 04 05	BAS	4	MVI	4	Same	12	These SETOF and COMP instructions will accomplish the same results as the Numeric compares on the previous page. This way will execute faster and takes 28-40 bytes less on Card or Tape. Takes MORE on Disk.
									MVI	4	MVI	4			
									MVI	4	MVI	4			
									MVI	4					
									BCR	2					
		setof						06 07	BAS	4					
									MVI	4	MVI	4	Same	8	
									MVI	4	MVI	4			
									BCR	2					

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper-ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4	Tape V2/L2	Disk V2/L0				
								Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes		
04 01	A	Add	B	B					CLI(04)	4	AP	6	AP	6	
									BC	4					
									CLI(01)	4					
									BC	4					
									AP	6					
										128		80		88	
METHOD A															
	A	Comp	0			=	=	99	BAS	4	Same	42	CP	6	METHODS OF COMPARING : Numeric COMP
									MVC	6	As		MVI	4	
									ZAP	6	Card		BC	4	
									ZAP	6			MVI	4	
									SP	6					
									BCR	2					
									MVI	4					
									BC	4					
									MVI	4					
	A	Comp	1					01		42		42		18	
	A	Comp	2					02		42		42		18	
	A	Comp	3					03		42		42		18	
	A	Comp	4					04		42		42		18	
	A	Comp	5					05		42		42		18	
	A	Comp	6					06		42		42		18	
	A	Comp	7					07		42		42		18	
										336		336		144	

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper- ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
<u>METHOD D</u>															
		setof						03 04 05	BAS	4	MVI	4	Same	12	These SETOF and Alpha COMP will accomplish the same results as the previous compares. This will execute the fastest of all examples. This way does take more core than the previous Alpha compare examples.
									MVI	4	MVI	4			
									MVI	4	MVI	4			
									MVI	4					
									BCR	2					
		setof						06 07	BAS	4	MVI	4	Same	8	
									MVI	4	MVI	4			
									MVI	4					
									BCR	2					
	G	Comp '00001'		=	Alpha			02 99 01	BAS	4	CLC	6	Same	42	
									CLC	6	MVI	4	As		
									BCR	2	BC	4	Tape		
									MVI	4	MVI	4			
									BC	4	MVI	4			
									MVI	4	BC	4			
									MVI	4	MVI	4			
									BC	4	MVI	4			
									MVI	4	BC	4			
									MVI	4	MVI	4			
									BC	4					
									MVI	4					
02	G	Comp '00003'						04 02 03		56		50		50	
04	G	Comp '00005'						06 04 05		56		50		50	
06	G	Comp '00007'						06 07		44		38		38	
										236		200	200		

RPG CALCULATION INSTRUCTIONS
CORE REQUIREMENTS

Condition Indicators	Fact. 1	Oper- ation	Fact. 2	Res. Fld.	Fld. Lth.	Dec. Pos.	Half Adj.	Result. Indicat.	Generated Instructions per Oper. System						Comments
									Card V2/L4		Tape V2/L2		Disk V2/L0		
									Instruct.	Bytes	Instruct.	Bytes	Instruct.	Bytes	
	A	Comp 1		=	=			02 99 01	BAS	4	Same	66	CP	6	
									MVC	6	As		MVI	4	
									ZAP	6	Card		BC	4	
									ZAP	6			MVI	4	
									SP	6			MVI	4	
									BCR	2			BC	4	
									MVI	4			MVI	4	
									BC	4			MVI	4	
									MVI	4			BC	4	
									MVI	4			MVI	4	
									BC	4					
									MVI	4					
									MVI	4					
									BC	4					
									MVI	4					
02	A	Comp 3						04 02 03		74		74		50	
04	A	Comp 5						06 04 05		74		74		50	
06	A	Comp 7						07 06		62		62		38	
										308		296	200		