

1	<p>Explain the following</p> <ul style="list-style-type: none"> A) Five Functions of an operating System B) Re-entrant code C) Memory Buffer Register D) Decimal Packed data format E) Logical (characters) Data Format F) Use of Index Registers G) Literals <p>Pseudo-ops</p>																																
2	<p>A) The following program computes the following</p> $A = 2 * B + 2 * C - 1$ <ul style="list-style-type: none"> 1) Compute START 2) USING *,15 3) L 1,B 4) SLA 1,1 5) L 2,C 6) SLA 2,1 7) AR 1,2 8) S 1,=F'1' 9) ST 1,A 10) BR 14 11) A DC F'0' 12) B DC F'5' 13) C DC F'7' 14) END <p>Verify that preceding program works correctly by simulating the instructions one by one and filling the table below</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 25%;">Instruction</th> <th style="width: 25%;">Register 1</th> <th style="width: 25%;">Register2</th> <th style="width: 25%;">Location A</th> </tr> </thead> <tbody> <tr><td>3</td><td></td><td></td><td></td></tr> <tr><td>4</td><td></td><td></td><td></td></tr> <tr><td>5</td><td></td><td></td><td></td></tr> <tr><td>6</td><td></td><td></td><td></td></tr> <tr><td>7</td><td></td><td></td><td></td></tr> <tr><td>8</td><td></td><td></td><td></td></tr> <tr><td>9</td><td></td><td></td><td></td></tr> </tbody> </table> <p>B) Assume STOMP is defined by</p> <pre>STOMP DC C'CERASURE'</pre> <p>How will the following instructions execute differently</p> <pre>MVC STOMP+1(8), STOMP MVC STOMP(8), STOMP + 1</pre> <p>C) What will be in register 3 after each instruction in the following sequence of instructions</p> <pre>LA 3, = A (XYZ) LR 3,3 L 3, = F'5' LA 3, 10(2,5)</pre>	Instruction	Register 1	Register2	Location A	3				4				5				6				7				8				9			
Instruction	Register 1	Register2	Location A																														
3																																	
4																																	
5																																	
6																																	
7																																	
8																																	
9																																	
3	<p>Make All the tables & Expand the Macro.</p>																																

```

MACRO
&LABEL EXITLINK &TYPE
AIF      (&TYPE' NE ").FOUND
MNOTE   '**** MISSING TYPE PARAMETER'
MEXIT
.FOUND ANOP
AIF      (&TYPE' EQ 'N').NORMAL
AIF      (&TYPE' EQ 'R').RETURN
AIF      (&TYPE' EQ 'V').VALUE
AIF      (&TYPE' EQ 'B').BOTH
MEXIT
.NORMAL ANOP
L        13,4(13)
LM       14,12,12(13)
BR       14
AGO      .DONE
.RETURN ANOP
L        13,4(13)
L        14,12(13)
LM       0,12,20(13)
BR       14
AGO      .DONE
.VALUE ANOP
L        13,4(13)
LM       14,15,12(13)
LM       1,12,24(13)
BR       14
AGO      .DONE
.BOTH ANOP
L        13,4(13)
L        14,12(13)
LM       1,12,24(13)
BR       14
.DONE ANOP
MEND
MAIN START
EXITLINK Q
EXITLINK
EXITLINK V
EXITLINK N
EXITLINK R
EXITLINK B
END MAIN

```

B) Explain the Standard System Linkage criteria of an IBM 370 assembler. Write the

minimum set of linkage instructions & then explain them one by one.

4 A)explain

- I. compile and Go Loaders
- II. Absolute Loaders
- III. Text Editors
- IV. Debuggers

B) For the following Loader tables generate the Assembly program as much as possible.

RLD

ID	Flag	Length	Relative address
01	+	4	40
03	+	4	56
02	-	4	64
01	+	4	64

TXT

Opcode	Relative address	Address Constant	Actual value
BALR	0		
SR	2		
L	4		
L	8		
ST	12		
BR	16		
DC	20		
DC	40	A(A+10)	66
DC	44		
DC	56	A(DELTA)	0
DC	60		
DC	64	A(STUDENT-SOLN)	0

ESD

Symbol	Type	ID	Relative address	Length
Student	SD	01	0	68
A	LD			
SOLN	ER	02		
DELTA	ER	03		

5 For the following Assembly Program :

```

PRGAM2  START  0
        USING  *,15
        LA    15,SETUP
        SR    TOTAL,TOTAL

AC      EQU    2
INDEX  EQU    3
TOTAL  EQU    4
    
```

```

DATABASE EQU 13
SETUP EQU *
        USING SETUP,15
        L DATABASE, = A(DATA1)
        USING DATAAREA,DATABASE
        SR INDEX, INDEX
LOOP L AC, DATA1 ( INDEX)
    AR TOTAL, AC
    A AC, = F'5'
    ST AC, SAVE (INDEX)
    A INDEX, = F'4'
    C INDEX, = F'8000'
    BNE LOOP
    LR 1,TOTAL
    BR 14
    LORG
SAVE DS 2000F
DATAAREA EQU *
DATA1 DC F'25,26,97,...2000 Nos
        END

```

Write

- a) Symbol Table
- b) Literal Table
- c) Base table

Machine Hex code (Take the hex opcode of every instruction as 'AB')