Computer Structure and Language

Hamid Sarbazi-Azad
Department of Computer Engineering
Sharif University of Technology (SUT)
Tehran, Iran

Example 1: Write an assembly program to add 100 elements of word array ARR and store it in SUM.

```
ADD     START   0

        Defining R12 as base register
        & initialize it to 6  → (R12) =6.

        LA      2,100     counter
        XR      3,3       index  = SR 3,3  = LA 3,0
        XR      4,4       accumulator  = SR 4,4 = LA 4,0 = LR 4,3
LOP     A       4,ARR(3)
        LA      3,4(3)
        BCT     2,LOP
        ST      4,SUM

        Returning to OS
SUM     DS      F
ARR     DS      100F
        END     ADD
```

Example 1+: Write an assembly program to add 100 elements of word array ARR and store it in SUM.

```
ADD     START   0

        Defining R12 as base register
        & initialize it to 6  → (R12) =6.

        LA      2,400           counter
        XR      4,4             accumulator
LOP     A       4,ARR-4(2)
        S       2,N4
        BNZ     LOP
        ST      4,SUM

        Returning to OS

ARR     DS      100F
SUM     DS      F
N4      DC      F'4'
        END     ADD
```


Example 1: Write an assembly program to add 100 elements of word array ARR and store it in SUM.

Address	Machine Code	Assembly Code			
000000		ADD	START	0	
		Defining R12 as base register & initialize it to 6 → (R12) =6.			Six Bytes
000006	41200064		LA	2,100	counter
00000A	1733		XR	3,3	index
00000C	1744		XR	4,4	accumulator
00000E	5A43C01E	LOP	A	4,ARR(3)	
000012	41330004		LA	3,4(3)	
000016	4620C008		BCT	2,LOP	
00001A	5040C1AE		ST	4,SUM	
00001E		Returning to OS			Six Bytes
000024		ARR	DS	100F	
0001B4	00000000	SUM	DC	F'0'	
0001B8			END	ADD	

Symbol Table

Symbol	B	Disp.
LOP	C	008h
ARR	C	01Eh
SUM	C	1AEh

Example 2: Write an assembly program to sort a100-element word array ARR in ascending order (Selection Sort).

```
SORT1  START  0

[Defining R12 as base register
 & initialize it to 6 → (R12) =6.]

       LA     2,99
       XR     6,6        outer index
LOP2   LA     7,4(6)     inner index
       LR     3,2
LOP1   L      4,ARR(6)
       L      5,ARR(7)
       CR     4,5
       BL     OUT
       ST     4,ARR(7)
       ST     5,ARR(6)
OUT    LA     7,4(7)
       BCT    3,LOP1
       LA     6,4(6)
       BCT    2,LOP2

[Returning to OS]

ARR    DS     100F
       END    SORT1
```


Example 2+: Write an assembly program to sort a100-element word array ARR in ascending order (Selection Sort).

```
SORT1  START  0

[Defining R12 as base register
 & initialize it to 6 → (R12) =6.]

       LA     2,ARR+396
       LA     6,ARR      outer pointer
LOP2   LA     7,4(6)     inner pointer
LOP1   L      4,0(6)
       C      4,0(7)
       BL     OUT
       L      5,0(7)
       ST     4,0(7)
       ST     5,0(6)
OUT    LA     7,4(7)
       CR     7,2
       BNP    LOP1
       LA     6,4(6)
       CR     6,2
       BL     LOP2

[Returning to OS]

ARR    DS     100F
       END    SORT1
```

Example 3: Write an assembly program to sort a100-element word array ARR in ascending order (Bubble Sort).

```
SORT2  START  0

[Defining R12 as base register
 & initialize it to 6 → (R12) =6.]

       LA     2,99
LOP2   XR     6,6
       LR     3,2
LOP1   L      4,ARR(6)
       L      5,ARR+4(6)
       CR     4,5
       BL     OUT
       ST     4,ARR+4(6)
       ST     5,ARR(6)
OUT    LA     6,4(6)
       BCT    3,LOP1
       BCT    2,LOP2

[Returning to OS]

ARR    DS     100F
       END    SORT2
```


Example 3+: Write an assembly program to sort a100-element word array ARR in ascending order (Bubble Sort).

```
SORT2  START  0

[Defining R12 as base register
 & initialize it to 6 → (R12) =6.]

       LA     6,ARR+396
LOP2   LA     3,ARR
LOP1   L      4,0(3)
       L      5,4(3)
       CR     4,5
       BL     OUT
       ST     4,4(3)
       ST     5,0(3)
OUT    LA     3,4(3)
       CR     3,6
       BNE    LOP1
       S      6,FOUR
       CR     2,6
       BNE    LOP2

[Returning to OS]

FOUR   DC     F'0'
ARR    DS     100F
       END    SORT2
```

Example 4: Write an assembly program to sort a100-element word array ARR in ascending order (Gnome Sort).

```
SORT3  START  0

[Defining R12 as base register
 & initialize it to 6 → (R12) =6.]

       LA     3,396
       XR     2,2          index
LOP    L      4,ARR(2)
       C      4,ARR+4(2)
       BC     B'1100',OUT  == BNH
       L      5,ARR+4(2)
       ST     4,ARR+4(2)
       ST     5,ARR(2)
       S      2,FOUR
       B      LOP
OUT    LA     2,4(2)
       CR     2,3
       BNE    LOP

[Returning to OS]
       DC     X'80000000'
ARR    DS     100F
FOUR   DC     F'4'
       END    SORT3
```


Example 4+: Write an assembly program to sort a100-element word array ARR in ascending order (Gnome Sort).

```
SORT3  START  0

[Defining R12 as base register
 & initialize it to 6 → (R12) =6.]

       LA     2,ARR
       LA     3,ARR+396
LOP    L      4,0(2)
       L      5,4(2)
       CR     4,5
       BNH    OUT
       ST     4,4(2)
       ST     5,0(2)
       S      2,EIGHT
OUT    LA     2,4(2)
       CR     2,3
       BNE    LOP

[Returning to OS]
       DC     X'80000000'
ARR    DS     100F
EIGHT  DC     F'8'
       END    SORT3
```

Example 4+: Write an assembly program to sort a100-element word array ARR in ascending order (Gnome Sort).

Address	Machine Code	Assembly Code			
000000		SORT3	START	0	
		Defining R12 as base register & initialize it to 6 → (R12) =6.			Six Bytes
000006	4120C036		LA	2,ARR	
00000A	4130C1C2		LA	3,ARR+396	
00000E	58420000	LOP	L	4,0(2)	
000012	58520004		L	5,4(2)	
000016	1945		CR	4,5	
000018	47C0C022		BNH	OUT	
00001C	50420004		ST	4,4(2)	
000020	50520000		ST	5,0(2)	
000024	5B20C1C6		S	2,EIGHT	
000028	41220004	OUT	LA	2,4(2)	
00002C	1923		CR	2,3	
00002E	4770C008		BNE	LOP	
000032		Returning to OS			Six Bytes
000038	80000000		DC	X'8000000'	
00003C		ARR	DS	100F	
0001CC	00000008	EIGHT	DC	F'8'	
			END	SORT3	

Symbol Table

Symbol	B	Disp.
LOP	C	008h
OUT	C	022h
ARR	C	036h
EIGHT	C	1C6h

End of Slides