Upmanyu Jha

8875

Te Comps A

SPCC Assign. 1

Q1. Design Pass1 of two Pass Assembler for given input source program.

sol.^{n} →

a. We know that, a symbol table is generated at end of pass 1 for a pass 2 assembler Therefore, after scanning the program from top to bottom every instruction is accounted for table that is generated.

b. Here, Pass 1 of the assembler helps to determine the size and address of data and instructions.

Therefore, for pass 1 we create an intermediate file that acts as a input to two pass.

c. Now, the given INPUT file:

```
INPUT
PG1        start      0
           USING    * , 15
             L      1 , FIVE
             A      1 , FOUR
             ST     1 , TEMP
             SR     2 , 2
             AR     1 , 2
           FIVE     DC     F'5'
           FOUR     DC     F'4'
           TEMP     DS     2F
           TEMP1    DS     3F
           END      PG1
```

Sol^n

INPUT :-

```
        PG1     0   START   0
                0   USING   *,15
                0   L       1, FIVE
                4   A       1, FOUR
                8   ST      1, TEMP
                12  SR      2, 2
                14  AR      1, 2
16  FIVE                DC      F'5'
20  FOUR                DC      F'4'
24  TEMP                DS      2F
32  TEMP1               DS      3F
44  END                 PG1
```

SYMBOL	LCValue	length	Relocation Absolute
PG1	0	1	R
FIVE	16	4	R
FOUR	20	4	R
TEMP	24	8	R
TEMP1	32	12	R

Q2. DESIGN Pass 2 of two pass assembler for the given input source Program.

→ a) In the pass 2 of the two pass assembler the objective code is generated by connecting symbolic opcode into respective numeric of code in MOT b) Here, the assembler stores all machine opcode with symbolic code, their length and their bit configuration.

It will also process ~~psed~~ pseudo - ops and will store them in POT (Pseudo opcode Table)

c) The databases required in Pass 2 assembler or Machine opcode Table, Pseudo opcode table, base table (for storing value of base register), location counter.

d) Given inpute file:

```
PG1       START
          USING  * 15        0
            L    1, FIVE     0          1, 1 16  (0, 15)
            A    1, FOUR     4          A, 1, 20 (0, 15)
           ST    1, TEMP     8          ST 1, 24
           SR    2, 2        12         SR 2, 2
           AR    1, 2        14         AR 1, 2
FIVE       DC       F'8'     16
FOUR       DC       F'4'     20
TEMP       DS       2F       24
TEMP1      DS       3F       32
           END               44
```

BASE REGISTER

	Availability Indicator 1 byte character	Designed relative address contents of base register (3 bytes = 24 bit address)
1	"N"	
2	"N"	
:		
15	"Y"	00 00 00

Q3. Using the above design give the output of pass1 and pass2 for the following code.

sol^n → Output for Pass 1 for the following would be as follows:

```
     PG2     START     0
0            USING     *, B
0            L         1, TWO
4            AR        1, Three
8            A         1, Three
10           M         1, =F'2'
14           ST        1, TEMP
18           SR        2, 2
20           B         EQU 4
40           TWO       DC H'2'
2            Three     DC P'2'
26           Temp      DS 1F
30           END
```

Symbol Table.

Symbol	LC Value	Length	Relocation Absolute
PG2	0	1	R
B	4	1	A
TWO	20	2	R
Three	22	4	R
Temp	26	4	R

Literal Table.

Literal	LC Value	Length	Relocate Absolute
=F'2'	30	4	R

2] Output for pass 2 for the following ~~would~~ would be as follows:-

Code :-

```
                START      0
                USING      *, B        0
                  L        1, TWO      ~~4~~0  L 1, 20(0,B)
                  AR       1, Three    4   AR 1, 22(0,B)
                  A        1, Three    6   A  1, 22(0,B)
                  M        1, =F'2'    10  30
                  ST       1, Temp     14  ST 1, 26
                  SR       2, 2        18  SR 2, 2
B                 EQU      4           20  —
Two               DC       H'2'        20  Two
Three             DC       F'2'        22  Three
Temp              DS       1F          26  Temp
                  END                  30
```

Base Register

	Available Indication 1 byte characters	Designed relative address contents of base register (3 byte = 24 bit address)
1	"N"	—
2	"N"	—
!	:	—
•	:	—
•		
15	"Y"	0000 00

Q4 Write the contents of symbol table and forward Reference table for the program given below:

```
RAM     START   0       0
        USING   *,15    0
          L     1,FIVE  0    L1, 12 (0,15)
          A     1,FOUR  4    A1, 12(0,15)
          ST    1, TEMP 8    ST 1,16( 0,15)
        DC      F'4'    12   DC
        DC      F'6'    14   DC
        DS      1F      16   DS
        END             20
```

Symbol table and forward reference table are generated after the Pass one of two pass assembler.
For all forward references mentioned in the program the assembly language generates a symbol table and literal table which help to specify the address of the operand.

Symbol	LC Value	Length	Relocation Absolute
RAM	0	1	R
FOUR	12	4	R
FIVE	14	4	R
TEMP	16	4	R

—x—