

Write, run, and analyze a program that adds 5 bytes of data and saves the result. The data should be the following hex numbers: 25, 12, 15, 1F, and 2B.

```

PAGE      60,132
TITLE     PROG2-1 (EXE)  PURPOSE: ADDS 5 BYTES OF DATA
.MODEL SMALL
.STACK 64
;
;-----
DATA_IN    .DATA
SUM         DB    25H,12H,15H,1FH,2BH
           DB    ?
;-----
MAIN       .CODE
PROC FAR
MOV AX,@DATA
MOV DS,AX
MOV CX,05           ;set up loop counter CX=5
MOV BX,OFFSET DATA_IN ;set up data pointer BX
MOV AL,0           ;initialize AL
AGAIN:      ADD AL,[BX] ;add next data item to AL
            INC BX      ;make BX point to next data item
            DEC CX      ;decrement loop counter
            JNZ AGAIN   ;jump if loop counter not zero
            MOV SUM,AL  ;load result into sum
            MOV AH,4CH  ;set up return
            INT 21H     ;return to DOS
MAIN       ENDP
END        MAIN

```

Write and run a program that transfers 6 bytes of data from memory locations with offset of 0010H to memory locations with offset of 0028H.

```

TITLE     PROG2-3 (EXE)  PURPOSE: TRANSFERS 6 BYTES OF DATA
PAGE      60,132
.MODEL SMALL
.STACK 64
.DATA
ORG 10H
DATA_IN    DB    25H,4FH,85H,1FH,2BH,0C4H
ORG 28H
COPY       DB    6 DUP(?)
;-----
MAIN       .CODE
PROC FAR
MOV AX,@DATA
MOV DS,AX
MOV SI,OFFSET DATA_IN ;SI points to data to be copied
MOV DI,OFFSET COPY     ;DI points to copy of data
MOV CX,06H             ;loop counter = 6
MOV_LOOP: MOV AL,[SI]   ;move the next byte from DATA area to AL
            MOV [DI],AL ;move the next byte to COPY area
            INC SI      ;increment DATA pointer
            INC DI      ;increment COPY pointer
            DEC CX      ;decrement LOOP counter
            JNZ MOV_LOOP ;jump if loop counter not zero
            MOV AH,4CH  ;set up to return
            INT 21H     ;return to DOS
MAIN       ENDP
END        MAIN

```

Write and run a program that adds four words of data and saves the result. The values will be 234DH, 1DE6H, 3BC7H, and 566AH. Use DEBUG to verify the sum is D364.

```

TITLE      PROG2-2 (EXE) PURPOSE: ADDS 4 WORDS OF DATA
PAGE      60,132
          .MODEL SMALL
          .STACK 64
;
;-----
DATA_IN    .DATA
           DW  234DH,1DE6H,3BC7H,566AH
           ORG  10H
SUM         DW  ?
;
;-----
MAIN       .CODE
           PROC FAR
           MOV  AX,@DATA
           MOV  DS,AX
           MOV  CX,04                ;set up loop counter CX=4
           MOV  DI,OFFSET DATA_IN  ;set up data pointer DI
           MOV  BX,00                ;initialize BX
ADD_LP:    ADD  BX,[DI]              ;add contents pointed at by [DI] to BX
           INC  DI                  ;increment DI twice
           INC  DI                  ;to point to next word
           DEC  CX                  ;decrement loop counter
           JNZ  ADD_LP              ;jump if loop counter not zero
           MOV  SI,OFFSET SUM        ;load pointer for sum
           MOV  [SI],BX              ;store in data segment
           MOV  AH,4CH               ;set up return
           INT  21H                 ;return to DOS
MAIN       ENDP
           END  MAIN

```

**Reference: THE 80X86 IBM PC AND COMPATIBLE COMPUTERS (VOLUMES 1 &2) ASSEMBLY LANGUAGE, DESIGN AND INTERFACING, Muhammed Ali Mazidi, Janice Gillispie Mazidi**