SOLUTIONS FOR CHAPTER 6

1. (a) 1110 1001 (b) 0000 1100 (c) 1101 1000 (d) 0110 1111
   (e) 1000 0000 (f) 0111 1111 (g) 0000 0001 0110 1101
   (h) 1000 0000 0000 0001

2. (a) OF = 0, result = 03 (b) OF = 1, result = 06
   (c) OF = 0, result = 47H (d) OF = 0, result = 00
   (e) OF = 0, result = 00

3. MOV AL,-122 ;AL = 86
   CBW ;AX = FF86
   MOV AX,-999H ;AX = F667
   CWD ;DX = FFFF
   MOV AL,17H ;AL = 17
   CBW ;AX = 0017
   MOV AL,127 ;AL = 7F
   CBW ;AX = 007F
   MOV AX,-129 ;AX = FF7F
   CWD ;DX = FFFF

4. The changes are highlighted below:
   (1) data name LOWEST was changed to HIGHEST
   (2) after the CMP, the conditional jump was changed to JGE

```
TITLE     PROB4  ;FIND THE HIGHEST TEMPERATURE
PAGE      60,132
;-----------------------------
STSEG     SEGMENT
          DB 64 DUP (?)
STSEG     ENDS
;-----------------------------
DTSEG     SEGMENT
SIGN_DAT  DB +13,-10,+19,+14,-18,-9,+12,-9,+16
          ORG 0010H
HIGHEST   DB ?
DTSEG     ENDS
;-----------------------------
CDSEG     SEGMENT
MAIN      PROC FAR
          ASSUME CS:CDSEG,DS:DTSEG,SS:STSEG
          MOV AX,DTSEG
          MOV DS,AX
          MOV CX,8                    ;load counter (number items - 1)
          MOV SI,OFFSET SIGN_DAT      ;set up pointer
          MOV AL,[SI]                 ;AL holds highest value found so far
BACK:     INC SI                      ;increment pointer
          CMP AL,[SI]                 ;compare next byte to highest
          JGE SEARCH                  ;IF AL is higher, continue search
          MOV AL,[SI]                 ;otherwise save new highest
SEARCH:   LOOP BACK                   ;loop if not finished
          MOV HIGHEST,AL              ;save highest temperature
          MOV AH,4CH
          INT 21H                     ;go back to DOS
MAIN      ENDP
CDSEG     ENDS
          END MAIN
```

5. STD is used to set the direction flag to one,
CLD is used to reset the direction flag to zero
the direction flag determines the direction of repeated string instructions, if DF = 0, the pointers (DI and SI) will be automatically incremented; if DF = 1, the pointers will be automatically decremented

6. (a) (b) (e)

7. (a) destination = DI, source = SI
(b) destination = DI, source = SI
(c) source = SI, destination = DI
(d) operand = SI, destination = AL
(e) operand = DI, source = AX
(f) operand = AX, destination = DI

8.

```
TITLE     PROB4  ;FIND THE HIGHEST TEMPERATURE
PAGE      60,132
;---------------------------
STSEG     SEGMENT
          DB 64 DUP (?)
STSEG     ENDS
;---------------------------
DTSEG     SEGMENT
DATA1          DB     'I pledge allegiance to the flag '
          DB     'of the United States of America,'
          DB     'and to the Republic for which it'
          DB     ' stands, one nation under God, indivisible, '
          DB     'with liberty and justice for all.  '
          DB     '************************!'
          ORG 300H
DATA2          DW     200 DUP (?)
DTSEG     ENDS
;---------------------------
CDSEG     SEGMENT
MAIN      PROC FAR
          ASSUME CS:CDSEG,DS:DTSEG,SS:STSEG,ES:DTSEG
          MOV  AX,DTSEG
          MOV  DS,AX                ;INITIALIZE THE DATA SEGMENT
          MOV  ES,AX                ;INITIALIZE THE EXTRA SEGMENT
          CLD                       ;CLEAR DIRECTION FLAG FOR AUTOINCREMENT
          MOV  SI,OFFSET DATA1      ;LOAD THE SOURCE POINTER
          MOV  DI,OFFSET DATA2      ;LOAD THE DESTINATION POINTER
          MOV  CX,200               ;LOAD THE COUNTER
          REP MOVSW                 ;REPEAT UNTIL CX BECOMES ZERO
          MOV AH,4CH
          INT 21H                   ;GO BACK TO DOS
MAIN      ENDP
CDSEG     ENDS
          END MAIN
```

9.

```
TITLE     PROB9
PAGE      60,132
;-------------------------
STSEG     SEGMENT
          DB 64 DUP (?)
STSEG     ENDS
;-------------------------
DTSEG     SEGMENT
ASC_DATA      DB    '0123456789'
          DB    '0123456789'
          DB    '0123456789'
          DB    '0123456789'
          DB    '0123456789'
          ORG 100H
COPY_DATA   DW     50 DUP (?)
DTSEG     ENDS
;-------------------------
CDSEG     SEGMENT
MAIN      PROC FAR
          ASSUME CS:CDSEG,DS:DTSEG,ES:DTSEG,SS:STSEG
          MOV AX,DTSEG
          MOV DS,AX
          MOV ES,AX
          CLD                          ;set up DF autoincrement
          MOV SI,OFFSET ASC_DATA       ;SI points to ASCII data
          MOV DI,OFFSET COPY_DATA      ;DI points to BCD data
          MOV CX,50                    ;50 bytes will be processed
LD_LOOP:  LODSB                        ;load an ASCII byte
          AND AL,0FH                   ;convert to unpacked BCD
          STOSB                        ;store unpacked BCD
          LOOP LD_LOOP                 ;continue looping
          MOV AH,4CH
          INT 21H                      ;go back to dos
MAIN      ENDP
CDSEG     ENDS
          END MAIN
```

10. REPNE

11.

```
TITLE     PROB11
PAGE      60,132
;-------------------------
STSEG     SEGMENT
          DB 64 DUP (?)
STSEG     ENDS
;-------------------------
DTSEG     SEGMENT
DATA_1        DB    'IbM'
DTSEG     ENDS
;-------------------------
```