



Remarks: (Answer the following questions... assume any missing data)

Question-1 (65 Marks)

The program written below used to copy the data block outlined in rectangle to the next column. Choose the best to complete the program

```

.model small
.stack 64
.data
.code
main proc
    mov ax,@data
    mov ds,ax
    mov cx,
    mov bx,3000h
again:
    mov ,ax
    dec
    jnz again
    mov ah,4ch
    int 21h
main endp
end main
    
```

1390:3000	49 00
1390:3010	20 00
1390:3020	4C 00
1390:3030	6F 00
1390:3040	76 00
1390:3050	65 00
1390:3060	20 00
1390:3070	6D 00
1390:3080	79 00
1390:3090	20 00
1390:30A0	63 00
1390:30B0	6F 00
1390:30C0	6C 00
1390:30D0	6C 00
1390:30E0	65 00
1390:30F0	67 00

- What is the function of the line labeled number 1 in the code
 - The definition of variables
 - Storage size of variables
 - Start of data section
 - all
 - The missing item labeled number 2 is
 - seg
 - reg
 - far
 - near
 - The missing item labeled number 3
 - 16H
 - 10H
 - 8H
 - 7H
 - The missing item labeled number 4
 - 3000H
 - [3000H]
 - BX
 - [BX]
 - The missing item labeled number 5
 - 30001
 - [3000]
 - [BX+1H]
 - [BX+10H]
 - The line missed in 6 is
 - INC BX,1H
 - INC BX,10H
 - ADD BX,10H
 - ADD BX,1H
 - The missing item labeled number 7
 - AX
 - BX
 - CX
 - None
 - for this program, the memory assigned to code segment and data segment are -----, -----
 - 64 Mbyte, 64 Mbyte
 - 64 Kbyte, 64 Mbyte
 - 64 Kbyte, 64 Kbyte
 - 64 Byte, 64 Byte
 - For this code (`MOV AX,A` `NEG AX` `ADD AX,5` `MOV A,AX`), it is equivalent to
 - $B = -A$
 - $B = 2*A+9$
 - $B = 5 - B$
 - $A = 5 - A$
- For the code (`MOV CL,C4` `ADD CL,A8`), After execution
- CF: carry flag is
 - 1
 - 0
 - ZF: zero flag is
 - 1
 - 0
 - SF: sign flag is
 - 1
 - 0
 - OF: Overflow flag is
 - 1
 - 0
 - PF: parity flag is
 - 1
 - 0
 - AF: auxiliary flag is
 - 1
 - 0
 - The SP register is typically used for accessing
 - strings
 - memory
 - stack
 - data segment

17. if the physical address of **?:14DAH** is **235DAH** what the value of the base address?
 A. 2110 B. 2210 C. 1022 D. 1220
18. To retrieve data from the stack we use -----
 A. pop B. push C. STO D. SAV
19. When we pushing data to stack, we always write the low byte of the stored value at the value of
 A. SP B. SP-2 C. SP-1 D. SP+2
20. A data segment is to be located from address **A0000** to **AFFFF**. What value must be loaded into DS?
 A. 000A B. AFFF C. FFFA D. A000
21. a dump command that will display the contents of the first 16 bytes of the current code segment is
 A. d cs:16 B. e cs:0000 16 C. d cs:0000 000f D. d cs:0000 16
22. The instruction **MOV [DI],DX** belongs to ----- addressing mode
 A. register B. based-relative C. Base-Plus-Index D. indirect
23. the instruction **ADD AX,[BX+SI]** belongs to ----- addressing mode
 A. register B. based-relative C. Base-Plus-Index D. indirect
24. in the command above, the two bytes memory contents at physical address ----- will be copied to AX
 A. CS:BX+SI B. DS:BX+SI C. DS:BX-SI D. CS:BX-SI
25. To define variable named cost as byte-sized in TASM with unknown value we use -----
 A. Cost DB ? B. DB Cost ? C. DB Cost = D. DW Cost =
26. To add line feed to your program output, you will write **MOV AH, 02** , **INT 21**, and put in DL -----
 A. 0dh B. odb C. 0ah D. 0af
27. To write **"I ENJOYED THE MICROPROCESSOR COURSE"** at **DS:0100**, you will write
 A. E 0100 " " B. D 0100 " " C. G 0100 " " D. T 0100 " "
28. **PUSH CS**
 A. Wrong B. Right
29. **POP CS**
 A. Wrong B. Right
30. All of these can be used as an option to *.model* except
 A. compact B. Moderate C. large D. small
31. the operating system need the entry point to the main program to be -----
 A. NEAR B. FAR C. BOTH D. NONE

Given that: DS = 7600H, SS = 6400H, BX = 7892H, BP = 1AF3H, and DI = 4572H

32. **MOV DX, [BX]**, the memory content that will be written to DX has the logical address
 A. 6400:1AF3 B. 7600:7892 C. 7892: 7600 D. 1AF3: 6400
33. **MOV BL, [BP]**, the memory content that will be written to BL has the logical address
 A. 6400:1AF3 B. 7600:7892 C. 7892: 7600 D. 1AF3: 6400
34. To get the negative of a number stored in DX, you will write -----
 A. NEG DX B. NOT DX C. XOR DX,8000h D. all
35. To get a character from the keyboard, call **INT 21H**, and put in AH -----
 A. 01H B. 02H C. 09H D. 21H

Regarding TASM: For the code shown next

36. Given that **NUMB = 16** (Decimal value), the program sequence will jump to
 A. Block1 B. Block2
37. Given that **NUMB = 17** (Decimal value), the program sequence will jump to
 A. Block1 B. Block2
38. Given that **NUMB = 18** (Decimal value), the program sequence will jump to
 A. Block1 B. Block2
39. To mask off all but bit 7 of the contents of the data register
 A. XOR DX,0080 B. OR DX,8000 C. AND DX,0080h D. AND DX,8000
40. To set all bits of an operand to 1, it could be ORed with -----
 A. FF B. 80 C. F0 D. 08
41. The **TEST** instructions works by performing a ----- operand on the operands and setting the flags
 A. OR B. XOR C. NOT D. AND
42. The relation between the old and new contents of AX after executing **NOT AX → ADD AX,1** is -----
 A. AX = NOT AX B. AX = AX - 1 C. AX = - AX D. None
43. If **AX = 070Dh**, what will be the content of DX after executing the **CWD**
 A. 0000h B. 070Dh C. FFFFh D. No change
44. If **AX = 070Dh**, what will be the content of AX after executing the **CWD**
 A. 0000h B. 070Dh C. FFFFh D. 070Ch

```

MOV CL,3
MOV AX,NUMB
DIV CL
CMP AH,0
JG block1
JMP block2

```

45. If **BX = 01A2h**, what will be the content of BX after executing **NEG BX**
 A. FE5Dh B. FE5Eh C. 5EFEh D. No change
46. If **AX = 0005h**, **BX = 0002h**, what will be the content of AX after executing **MUL BX**
 A. 0010h B. 000Ah C. 0000h D. No change
47. If **AX = ??05h**, **factor = FFh**, what will be the content of AX after executing **MUL factor**
 A. FFFBh B. FB04h C. 04FBh D. 0000h
48. If **AX = ??05h**, **factor = FFh**, what will be the content of AX after executing **IMUL factor**
 A. FFFBh B. FB04h C. 04FBh D. 0000h
49. If **AX = 0064h**, **divisor = 0Dh**, what will be the content of AX after executing **DIV divisor**
 A. F123h B. 0907h C. 0064h D. 0000h
50. If **DX = 0000h**, **AX = 0064h**, **CX = FFF3**, what will be the content of AX after executing **IDIV CX**
 A. 0009h B. 0900h C. FFF9h D. F9FFh
51. If **DX = 0000h**, **AX = 0064h**, **CX = FFF3**, what will be the content of DX after executing **IDIV CX**
 A. 0009h B. 0900h C. FFF9h D. F9FFh

```

-r
AX=0000  BX=0000  CX=0017  DX=0000  SP=0040  BP=0000  SI=0000  DI=0000
DS=141A  ES=141A  SS=142C  CS=142A  IP=0000  NV UP EI PL NZ NA PO NC
                B82C14          MOV     AX, 142C
  
```

what is the value written here ?

- During working with DEBUG, we have the print out shown above
52. What is the value written in the section pointed to by the arrow?
 A. Flags B. Logical address C. Segment register D. Machine code
53. What is the size of this file?
 A. 23 bytes B. 141A bytes C. 142C bytes D. SI bytes
54. The flag section shows that there is ----- happened
 A. negative B. carry C. parity even D. no auxiliary carry
55. the next instruction that will be executed has the machine code -----
 A. 000000 B. 141A C. one cannot know D. B82C14
56. The memory content of 142A:0000 to the next 2 bytes is -----
 A. 000000 B. 141A C. one cannot know D. B82C14
- The original contents of AX, BL, word-size memory location SUM, and carry flag (CF) are 1234h, ABh, 00CDh, and 0h, respectively. Describe the results of executing the following sequence of instructions:
ADD AX, [SUM] ADC BL, 05H INC WORD PTR [SUM]
57. AX = -----
 A. 1234h B. 1301h C. 01FE D. CC12
58. BL = -----
 A. ABh B. B0h C. 1Fh D. B8
59. SUM = -----
 A. 1234h B. 00CEh C. C6FE D. 5050
60. The mnemonics for adding to the accumulator the contents of memory location with offset address 0025
 A. ADD AX,[0025H] B. INC [0025] C. ADD AX,0025H D. INC 0025H
61. If **AX = 7fffH**, **BX = 8000**, and the code **CMP AX,BX JA below** is executed, the program transfer to label **below**
 A. RIGHT B. WRONG
62. The code **MOV CL,NUMB BT CL,1** can be used to
 A. Test the LSB of NUMB B. check odd or even C. Both A and B D. NONE
63. The output of **BTR CL,4** will be saved in
 A. CL B. ZF C. CH D. CF
64. when executed will mask off all but bit 7 of the contents of the data register
 A. AND DX,0080 B. XOR DX,0080 C. OR DX,0080 D. BT DX,0080
65. Write an instruction that will subtract the word contents of the storage location pointed to by the base register BX and the carry flag from the accumulator
 A. SBB AX,[BX] B. SUB AX,[BX] C. SUB AX,BX D. DEC AX,[BX]

Question-2 (25 Marks)

- A. Write a program in assembly code to count the characters in a line
- B. Write the core of assembly program to read in characters that end with blank "space" character

Q	A	B	C	D
1.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
12.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
14.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
15.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
16.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
17.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
18.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
19.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
20.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Q	A	B	C	D
21.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
22.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
23.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
24.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
25.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
26.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
27.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
28.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
29.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
30.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
31.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
32.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
33.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
34.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
35.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
36.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
37.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
38.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
39.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
40.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Q	A	B	C	D
41.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
42.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
43.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
44.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
45.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
46.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
47.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
48.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
49.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
50.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
51.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
52.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
53.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
54.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
55.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
56.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
57.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
58.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
59.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
60.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
61.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
62.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
63.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
64.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
65.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Best wishes
Dr. Mostafa A. Elhosseini