

Mid Term Assignment for C++

Ruhul Amin
18-38001-2

①	Name	status	②	Number	status
	Two words	— legal		246	— decimal
	?1	— legal		246h	— hex
	Two words	— illegal		1001	— decimal
	.@	— illegal		1,001	— illegal
	\$145	— legal		243h	— hex
	let's go	— illegal		FFFEh	— illegal
	T=	— illegal		0Ah	— hex
				Bh	— illegal
				1100B	— binary

③ Ans:

- a) A DW 52
- b) WORD1 DW ?
- c) DB 25h
- d) c1 DB ?
- e) WORD1 DW 65536
- f) ARRAY1 DW 1,2,3,4,5
- g) BELL EQU 02h
- h) MSG EQU 'THIS A MESSAGE'

Ruhul Amin
18-58001-2

Variable	Countent	Address
A	7	0000
B	Be	0001
C	1A	0002
	14	0003
	E	0004
	L	0005
	L	0006
	0	0007

A: 0000 B: 0001 C: 0003

1A

E

0007

③

Instruction

Status

Ruhul Amin
18-38001-2

MOV DS, AX

legal

MOV DS, 1008h

illegal ; can't use seg register with an immediate variable

MOV CS, ES

illegal ; segment register can't go together

MOV W1, DS

legal

XCHG W1, W2

illegal

SUB 6, B1

illegal

ADD B1, B2

illegal

ADD AL, 256

illegal

MOV W1, B1

illegal

Q

a) $A = B - A$

MOV AX, a

MOV BX, b

SUB BX, AX

MOV AX, BX

b) $A = -(A+1)$

MOV AX, A

ADD AX, 1

NEG AX

MOV a, AX

c) $e = A + B$

MOV AX, a

ADD AX, b

MOV e, AX

d) $B = 3 * B + 7$

MOV AX, B

ADD AX, B

ADD AX, B

ADD AX, 7

MOV B, AX

e) $A = B - A - 1$

MOV BX, b

SUB BX, a

SUB BX, 1

MOV a, BX

Ruhul Amin
18-38001-2

Q2

a) `mov ah, 1`
`int 21h`
`mov ah, 2`
`mov dl, al`
`int 21h`

b) `mov ah, 1`
`int 21h`
`add al, 20h`
`mov ah, 2`
`mov dl, al`
`int 21h`

Q3 a) code write :

• model small

• stack 100h

• data

`msg db 0Ah, 0Dh, "The sum of 5"`

`var db ?, 'and'`

`var2 db ?, ' is '`

`sum db ?, "5"`

• code

main proc

~~data~~

`mov ax, @data`

`mov ds, ax`

; initialize data seg to code segment

Rahul Arora
18-38001

; Display ? to prompt the user for input

mov ah, 2

~~mov dx, ax~~

mov al, ?

int 21h

; Read decimal digits

mov ah, 1

int 21h

mov var, al ; storing var another register

; Input 2nd variable

add al, var

sub al, 30h ; there is ascii coding in background so subtracting

mov sum, al

; Display msg

leq dx, msg

mov ah, 9

int 21h

exit:

mov ah, 4ch

int 21h

main endp

end main

Code writing:

Rahul Amin
18-38001-2

- model small
- stack 100h
- data

msg db 0Ah, 0Dh, "Enter three initials: \$"

a db ?

b db ?

c db ?

• code

main proc

mov ax, @data

mov ds, ax

; Display lines

mov ah, 9

lea dx, msg

int 21h

; Tracking input

mov ah, 1 ; 1st input

int 21h

mov al, al

mov ah, 2 ; 2nd input

int 21h

mov al, al

mov ah, 3 ; 3rd input

int 21h

mov al, al

; Line field

mov ah, 2

mov al, 0Ah

int 21h

mov al, 0Dh

int 21h

; Display function

mov ah, 2

mov dl, a

int 21h

; line field

mov ah, 2

mov dl, 0Ah

int 21h

mov dl, 0Ah

int 21h

; Display 2nd ~~and~~ number

mov ah, 2

mov dl, b

int 21h

; line field

mov ah, 2

mov dl, 0Ah

int 21h

mov dl, 0Ah

int 21h

; Display

mov ah, 2

mov dl, c

int 21h

; Das exit

exit:

mov ah, 4Ch

int 21h

main endp

end main

Rahul Amin
18-380012

10) Code writing :

Ruhul Amin
18-38601-2

- model small
- stack 10sh
- data

msg db "Enter a hex digit" ~~char~~ , \$

c db ?

msg2 db 0Ah, 0Ah, "In decimal it is: \$"

c2 db ?

main proc

mov ax, @data

mov ds, ax

; displaying first message

mov ah, 9

lea dx, msg

int 21h

; input

mov ah, 1

int 21h

mov c, al

; calculation

sub al, 11h

mov c2, al

; display output

mov ah, 9

lea dx, msg2

int 21h

exit;

mov ah, 4ch

int 21h

main endp

end main

Ruhul Amin
18-38001-2

pure segment

exit:

mov ah, 94ch

int 21h

main endp

end main

12) Code writing

- model small
- stack 100h
- data

msg db "9 \$"

msg1 db 0Ah, 0Ah, "***** \$"

msg2 db, 0Ah, 0Ah, "*** \$"

ch2 db ?

ch2 db ?

ch3 db ?, "*** \$"

main Proc

mov ax, @data ; Initialize data seg to code segment
mov ds, ax

; Prompt the user for ?

mov ah, 2
mov dl, msg
int 21h

; Enter initials

mov ah, 1
int 21h
mov ch, al
int 21h

mov ch2, al
int 21h
mov ch3, al

; output

mov ah, 9

lea dx, msg1
int 21h

...

int 21h

~~lea~~ lea dx, msg2
int 21h

lea dx, msg1

int 21h

...

...

int 21h

; Beep the counter

mov ah, 2
mov dl, 7h
int 21h

exit:

mov ah, 4ch

int 21h

main endp

end main