

Unit IV: Assembly Language Programming

METRO

HOLLOW CRAFT STICKER

Centre No.:
केंद्र क्रमांक

Seat No.
बैंटक क्रमांक
(in Figure)
(अंकांक)

(in Words)
(अंकांक)

Candidate's
Signature
परीक्षार्थीची सही

Supervisor's
Signature
पर्यावरणाची सही

बारकोड ह्या चौकटीत चिकटवा.

Paste the Barcode Sticker here

Hollow Craft Sticker

Weightage

20 MARKS

Hollow Craft Sticker

SSC HSC

फक्त UID कोडसाठी जागा

(परीक्षक वापरासाठी)

SSC HSC

Subject / विषय :	
Paper पेपर :	
Date दिनांक :	
Language of Answer / उत्तर लेखनाची भाषा :	

Supplements attached

Main Answer Book	No. of supplements	Total in Figure
1	+	=

Specific remarks of centre conductor regarding malpractices (in Red Ink)

उत्तरांकांनुसार प्रदर्शनाचा मोरक्कामांगंदर्दी कर्वा
मध्यांकांचे अंदांक (मात्र शाळांमध्ये)

	Marks in Figure	Marks in Words	Signature	Appl. No.
Examiner				1043
Moderator				1044
Chief Moderator				

बारकोड संबंधी सूचना

- विद्यार्थ्यानी प्रथम प्रविष्टकाकडून बारकोड स्टिकर घ्यावे.
- बारकोड वरीत बैंटक क्रमांक व विषय वरोवर असल्याची खाली करून घ्यावी.
- बारकोड स्टिकर त्यासाठी दिलेल्या जागेगांधे घर्डी पूऱ्याने देता चिकटवावा.
- बारकोडवर अन्य कोणतेही लिखाण करू नये. केल्यास परीक्षेतील गैरमार्गाचा प्रकार मानव्यात येईल.
- उत्तर प्रक्रिकेच्या मुद्दपृष्ठदरील अन्य माहिती, विहित जागेत लिहावी.

Q. No.	Examiner	Moderator	Chief Moderator
1	1		
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
Total in Figure			
Total in Words			
Signature			
Appl. No.			

Q.No.

Q.No.

MCQ

4.1 Model of Assembly Language Programming.

प्रश्नक.
Q.No.

→ Instructions, labels, symbols, etc.

• directive

Data Segment

Var1 DB ---

Var2 DB ---

Const EQU ---

< Data Segment of Program >

Data ends.

Code Segment

Assume CS:CODE, DS:Data

Start:

Instruction

Code Ends

End Start.

• model SMALL

Var1 DB

Var2 DW

Const EQU ---

• code

• Start UP

Start:

• EXIT

26. END

8 Programs

Q.No.							
Q.No.							

Addition of two, 8 bit numbers

Q1

Data Segment

a db 20h

b db 30h

c dw ?

Data Ends

Code Segment

Assume cs:code, ds:Data.

Start: mov AX, Data
 mov DS, AX
 mov AL, a
 mov BL, b
 add AL, BL
 mov C, AX
 int 3

Code Ends

End Start

Q.No.							
Q.No.							

Subtraction of two, 8 bit numbers

Q2

Data Segment

a db 30h

b db 20h

c dw ?

Data Ends

Code Segment

Assume cs:code, ds:Data.

Start: mov AX, Data
 mov DS, AX
 mov AL, 90h
 mov BL, 90h
 sub AL, BL
 mov AX, C
 int 3

Code Ends

End Start

Q No.							
Q No.							

Q2 Addition of 2, 16bit. Numbers

Data Segment
 a dw 2020h
 b dw 1000h
 c dd ?
 Data Ends.

Code Segment
 Assume CS:code, DS:Data
 Start: mov AX, Data
 mov DS, AX
 mov AX, a
 mov BX, b
 add AX, BX
 mov CX, AX
 int 3

Code Ends
 End start.

Q No.						
Q No.						

Q3 Subtraction of 2, 16bit. Numbers

Data Segment
 a dw 2020h
 b dw 1000h
 c dd ?
 Data Ends.

Code Segment
 Assume CS:code, DS:Data
 Start: mov AX, Data
 mov DS, AX
 mov AX, a
 mov BX, b
 sub AX, BX
 mov CX, AX

int 3
 Code Ends
 End start.

unsigned of
signed -

Q.No.										Q.No.								
-------	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--

प्र० नं.
Q.No.

Q.S

Multiplication of two 8bit or 16bit numbers. (unsigned)

Data Segment

a dw 2020h ; 00101000b ; 8bit db 120h
b dw 1010h ; 00001010b db 20h
c dd ? ; 00000000h

Data Ends.

Code Segment

Assume CS:Code, DS:Data, DS:[bx]

Start : mov ax, DS

mov DS, AX

mov BX, AX ; al

mov BX, AB ; bl

mul BX, AX ; bl

mov C, AX

int 3

Code Ends. Int 3 by R.O.J

End Start.

Q6. ~~Division of two 8 bit or 16 bit number (unsigned numbers)~~

Data Segment

a dw 2020h

b dw 1010h

c dd ?

Data Ends

Code Segment

Start: assume cs:Code, ds:Data

Start: mov AX, DCH ; tool2

mov DS, AX

mov AX, 29H

mov BX, 10H

DIV BX ; tool3

mov ECX, AX

int3

Code Endl

End Start.

Q.No.								Q.No.				
-------	--	--	--	--	--	--	--	-------	--	--	--	--

Q.No.

~~Q7. multiplication of 8-bit or 16bit two signed numbers using addressing~~

Data Segment.

a dw 1020h

b dw 2020h

c dd ?

Data Ends.

Code Segment

Assume CS:code, DS:Data

Start:

mov AX, Data

mov DS, AX

mov AX, 1000h

mov BX, b

imul BX

mov C, AX

int 3

Code Ends

End start

Do not write your name or seat no. below this line

Q.No.

Q.No.

Q.No.

Division of 2 (8-bit or 16-bit)

~~Q8~~

numbers (signed) for

Data Segment with words

a $\text{d}w\ 1234h$

b $\text{dw}\ 1020h$

c $\text{dd}\ 5\ 1010\ 3000\ 209$

Data Ends

$2000\ 0000$

Code Segment

Assume CS:CODE DS:DATA

Start: MOV AX, DATA

Mov DS, AX

XOR BX, BX

Mov AX, BX

Div BX

Output: MOV CX, AX

INT 3

Code Ends

End Start

1A 3H

T1.10001 9H

4C 241 2 00000000

12 021 2 00000000

12 021

T2.10001

12 021 021 0000

12 021 021 0000

12 021 021 0000

12 021 021 0000

Do not write your name or seat no. below this line

Q.No.						Q.No.				
-------	--	--	--	--	--	-------	--	--	--	--

प्राक्त.
Q.No.

(09)

Counting Positive and Negative number
in an Array (i.e Block) of Deter.

→ Data Segment

Array dw F423H, 6523H, B658H, 1232H,
1333H, 1444H, ...

Count dw 06H nbt d

Pos_Count dw ? nbt d

Neg_Count dw ? nbt d

Data End

End

Code Segment

Assume CS:Code, DS:Deter

Start: Mov AX, DS

MOV DS, AX

Mov DX, 10000H

Mov CX, Count

Mov SI, Offset Array

NEXT: Mov AX, [SI]

ROR AX, DH

JC Negative_in_Bit

INC DL

Jump COUNT_IT

Negative : INC DH

COUNT_IT : INC SI

INC SI

LOOP NEXT

Mov Neg_Count, DL

Mov Pos_Count, DH

Mov AH, 4CH

INT 21H

Code End

End Start

Q.No.

Q.No.

Q.No.

~~Q10. Counting Even and odd numbers in an array (i.e. Block) of Data.~~

Data Segment
 num: dw 4444h, 5555h, 2222h, 7777h,
 1111h, 3333h, 8888h, 6666h

Count dw 0008h ; begin
 Odd count db ?
 Even count db ?

Data Ends

Code Segments

Assume cs:code, ds:Data.

Start: mov ax, data

mov DS, AX

mov DX, 0008h

mov CX, count

mov SI, offset num

again: mov AX, [SI]

RB AX, 01h

JC Odd

inc AX

jmp common

Odd: inc DH

common: inc BX

loop again

mov Odd, count, DH

mov Pos, count, DL

Code End

End Start.

Q1 ALP to find largest no in Array.

(Q1) ALP to find Largest no in Array.

Data Segment

nums dw 4444h, 5555h, 6666h, 7777h
1111h, 3333h, 4444h, 3333h

counts dw 0008h

Largest dw ?

Date Ends.

Code Segment

assume cs:Code, ds:Data, ss:Stack

start: mov ax, ds

mov bx, offset nums

mov cx, counts

mov bx, offset nums

again: cmp ax, [bx]

inc skip

mov ax, [bx]

skip: inc bx

inc bx

loop again

mov largest, ax

Code ends. db 0Dh, 0Ah, 0Dh, 0Ah
int 21h, retf 0Dh, 0Ah, 0Dh, 0Ah

Code ends. db 0Dh

Code End Start, var

db 0Dh, 0Ah, 0Dh, 0Ah
db 0Dh, 0Ah, 0Dh, 0Ah

dw 0008h

Q2 ALP to find smallest no in Array

(Q2)

Data Segment

nums dw 4444h, 5555h, 6666h, 7777h
1111h, 3333h, 4444h, 3333h

count dw 0008h

Smallest dw ?

Date Ends.

start: Code Segment

assume cs:Code, ds:Data, ss:Stack

start: mov ax, ds

mov bx, offset nums

mov cx, count

mov bx, offset nums

again: cmp ax, [bx]

jle skip

skip: inc bx

inc bx

loop again

mov smallest, ax

int 21h

Code ends. db 0Dh, 0Ah, 0Dh, 0Ah

End start. db 0Dh, 0Ah, 0Dh, 0Ah

db 0Dh, 0Ah, 0Dh, 0Ah

dw 0008h

db 0Dh, 0Ah, 0Dh, 0Ah

db 0Dh, 0Ah, 0Dh, 0Ah

dw 0008h

db 0Dh, 0Ah, 0Dh, 0Ah

db 0Dh, 0Ah, 0Dh, 0Ah

dw 0008h

Q No.						
Q No.						

Q13 ALP to sort numbers in Ascending order.

Data Segment

String1 DB 99h, 12h, 56h, 45h, 36h
Data Ende

Code Segment

Assume CS:Code, DS:Data, SS:

Start: MOV AX, Data

MOV DS, AX

MOV CX, 05H

MOV SI, 00H

UP2: MOV CL, 05H

LEA SI, String1

UP1: MOV AL, [SI]

MOV BL, [SI+1]

CMP AL, BL

JC B Down

MOV DL, [SI+1]

XCHG [SI+1], DL

MOV [SI+1], DL

Down: INC SI

DEC CL

JNZ UP1

DEC CL

JNZ UP2

INT3

Code Ende

End Start.

Q No.						
Q No.						

Q13 ALP to Sort numbers in Descending Order.

Data Segment

String1 DB 99h, 12h, 56h, 45h, 36h
Data Ende

Code Segment

Assume CS:Code, DS:Data

Start: MOV AX, Data

MOV DS, AX

MOV CH, 05H

UP2: MOV CL, 05H

LEA SI, String1

UP1: MOV AL, [SI]

MOV BL, [SI+1]

CMP AL, BL

JNC Down

MOV DL, [SI+1]

XCHG [SI+1], DL

MOV [SI+1], DL

Down: INC SI

DEC CL

JNZ UP1

DEC CL

JNZ UP2

INT3

Code Ende

End Start.

O.No.								O.No.				
-------	--	--	--	--	--	--	--	-------	--	--	--	--

O.No.

DATA
CNo
59

~~Q15 Write ALP for addition of two 16 bit BCD Numbers~~

A

⇒ Data Segment

N1 DW 2804H ; bcd 0010000000000100

N2 DW 1010H

BCD sum DW -

Data EndS

Code Segment

Assume CS:Code, DS:Data

Start = mov AX, Data

1. mov DS, DX ; L900

2. mov AX, N1

mov BX, N2

3. Add AL, BX ; L901

4. mov AL, 40H ; L902

mov CL, AL

mov AL, 10H

5. Add AL, DS[BX] ; L903

6. mov AL, DS[BX]

7. mov CX, AL

mov BCD sum, CX

8. INT 10H ; L904

INT24H ; L905

Code EndS ; L906

End Start ; L907

SEC
ENC
CON

8.111
Data section
Sum2.bas

Q.No.									Q.No.				
-------	--	--	--	--	--	--	--	--	-------	--	--	--	--

प्रश्नक.
Q.No.

Q16

Write ALP for subtraction of two
16-bit BCD numbers

Data Segment:

N1 DW 2804H

N2 DW 2234H

Register ends:

A BX D N3

Code Segment:

Assume CS:Code, DS:Data

Start: MOV AX, DS

MOV AX, N1

MUL AX, \$10000H

XCHG AX, N2

Sub AL, BL

DAS

MOV CL, AL

MOV AL, AH

SBB AL, BH

DAS

MOV CH, AL

SHL CL, 4

MOV BX, CL

INT 21H

Code Ends

End Start.

Q.No.

Q.No.

Q.No.

Q.F

Find the length of string

Data Segment

Str DB 'Good morning'

LEN DB ?

Data Ends

Code Segment

Assume CS:Code, DS:Data

Start:

Mov AX, DS

LEA SI, Str

Mov CL, 00H

Mov AL, [SI]

NEXT: Cmp AL, '\$'

JZ EXIT

Add CL, 01H

Inc SI

JMP

NEXT: EXIT: Mov LEN, CL

WT211

Code Ends

End Start

Do not write your name or seat no. below this line

O.No.

O.No.

Q.No.

Q18

Program for sum of series

Data Segment

DataBlock DB 45h, 34h, 36h, 37h,
4f3fh, 55h, 60h, 70h, 80h, 90h

ResultLSB DB @ 001

Result_msb DB '0'

Data_Ends DW 0000h

?

Jump to start

Code Segment

Assume CS:Code, DS:Data

Start: MOV AX, Data

MOV BX, DataBlock

MOV CL, 0AH

UP: MOV AL, [BX+CS:SI]

ADD RESULT, AL, ES:[SI]

JNC DOWN

INC RESULT, MSB

DOWN: INC SI, MSB

LOOP UP INT 30H

Code Endl JNC : DOWN

End Start SHT

AL = 00H

XL = 00H

CL = 0AH

CS:SI = 0000H

bx = 0000H

Do not write your name or seat no. below this line

Q.No.					Q.No.						
-------	--	--	--	--	-------	--	--	--	--	--	--

प्रश्न संख्या
Q.No. 019 Count 0's & 1's from 16 bit
Number

Ans : Data segment

1. DS DW 25648H

2. DS DW 0000H

③ DW 0000H

Data Ends

Code Segment

Assume DS: Data, CS: Code

Start: mov AX, Data

mov CX, 1000H

mov BX, 0000H

mov DX, 0000H

UP: ROL AX, 1H

JC ONE

INC BX

JMP NEXT

ONE: INC DX

NEXT: DEC CX

JNZ UP

mov Z, BX

mov O, DX

INT3

Code Ends

End Start

Q.No.

Q.No.

Q.No.
Q.No.

Q20

Write ALP to transfer 10 bytes of data from one memory location to another, also draw flowchart for same.



Data Segment

String1 DB 20h, 02h, 03h, 04h, 05h
06h, 07h, 08h, 09h, 0Ah

String2 DB 10 DUP(0)

Data End

String2

Code Segment

max 8 sum exec code, DS: Data

Start: mov AX, Data

mov DS, AX

mov ES, AX

LEA SI, String1

LEA DI, String2

mov CX, 000A

CLD

REP MOVS B

INT 3

Code End

End Start

GOTB

Q No.						Q.No.		
-------	--	--	--	--	--	-------	--	--

Flowchart

Start

→ Jump to block 1

Load DS register with A value

Load ES register with B value

A value

→ Jump to block 2

Load SI register with offset
Address of string 1 loaded

Load DI register with offset

Address of string 2 loaded

Initialize counter with 0

CX

0.00

Set direction flag, PL

DL

0.00

Repeat string move loop

SI

0.00

DI

0.00

CX

0.00

PL

0.00

SI

0.00

DI

0.00

CX

Q.No.						Q.No.			
-------	--	--	--	--	--	-------	--	--	--

~~(Q2)~~ Reverse a string in R (both directions).
flowchart for same.

→ Data Segment

str DB 'Good morning \$'
REY DB OFH DUP(?)

Data ends

Code ends

Code Segment

Start? Assume DS:Data, CS:Code

MOV AX, 200011H

MOSI, DS, BX, SI, DI

LEA SI, str

MOV CX, OFH

LEA DX, REY

ADD DI, OFH

UP: MOV AL, [SI]

MOV [DI], AL

INC SI

Start

DEC DI

LOOP UP

INT 2FH [IA] →

Code ends

End Start

SI ← Address of string

DI = CX + DI

SI ← DI - AL

DI ← DI - AL

No DI = DI - 1

O.No.										O.No.				
-------	--	--	--	--	--	--	--	--	--	-------	--	--	--	--

PFB,
O.No.

Concat two strings

- Model small

- Data

Str1 DB 'Hello'

Str2 DB 'World'

- Code

Mov Ax, @Data

Mov DS, AX

Mov SI, offset Str1

Next: mov AL, [SI]

Cmp AL, '\$'

JE Exit

Inc SI

Jmp Next

Exit?: Mov DI, offset Str2

Up: Mov AL, [DI]

Cmp AL, '\$'

JE Exit 1

Mov [SI], AL

Inc SI

Inc DI

Jmp Up

Exit 1: Mov AL, '\$'

Mov [SI], AL

WT 214

Ends

End.