

I N D E X

1910020362

NAME: OMKAR SAVANT

ROLL NO.: 45

STD.: _____ DIV./SEC.: S YCM SUBJECT: 72415 - Microprocessor
- Manual

Content Page
List of Practicals and Progressive Assessment Sheet

Sr. No.	Title of the practical	Page No.	Date of performance	Date of submission	Assessment marks (50)	Dated sign. of teacher	Remarks (If any)
1.	Identify the various pins of the given microprocessor	1					
2.	Use the assembly language programming tools and functions	6					
3.	Use different addressing mode instruction in program (a) Write an Assembly Language Program (ALP) to add two given 8 and 16 bit numbers. (b) Write an Assembly Language Program (ALP) to subtract two given 8 and 16 bit numbers.	12					
4.	(a) Write an ALP to multiply two given 8 and 16 bit unsigned numbers. (b) Write an ALP to multiply two given 8 and 16 bit signed numbers.	20					
5.	(a) Write an ALP to perform block transfer data using string instructions (b) Write an ALP to perform block transfer data without using string instructions.	27					
6.	(a) Write an ALP to compare two strings without using string instructions. (b) Write an ALP to compare two strings using string instructions	33					
7.	(a) Write an ALP to divide two unsigned numbers (b) Write an ALP to divide two signed numbers	39					
8.	Write an ALP to add, subtract, multiply, divide two BCD numbers.	47					
9.	(a) Write an ALP to find sum of series of Hexadecimal Numbers. (b) Write an ALP to find sum	54					

Sr. No.	Title of the practical	Page No.	Date of performance	Date of submission	Assessment marks (50)	Dated sign. of teacher	Remarks (if any)
	of series of BCD numbers.						
10.	Write an ALP to find smallest number from array of n numbers. Write an ALP to find largest number from array of n numbers.	61					
11.	(a) Write an ALP to arrange numbers in array in ascending order. (b) Write an ALP to arrange numbers in array in descending order.	68					
12.	(a) Write an ALP to arrange string in reverse order (b) Write an ALP to find string length. (c) Write an ALP to concatenation of two strings.	74					
13.	(a) Write an ALP to check a given number is ODD or EVEN. (b) Write an ALP to count ODD and/or EVEN numbers in array.	80					
14.	(a) Write an ALP to check a given number is POSITIVE or NEGATIVE (b) Write an ALP to count POSITIVE and/or NEGATIVE numbers in array.	86					
15.	(a) Write an ALP to count number of '1' in a given number (b) Write an ALP to count number of '0' in a given number	92					
16.	An assembly language program using procedures (a) Write an ALP for addition, subtraction, multiplication	98					

Date: 1/1

Saathie

1. Identify the Various Pins of 8086 microprocessor

* Observations :

Label Pins of 8086 in respective mode

		M1N MODE	MAX MODE
GND	1	40	Vcc
AD ₁₅	2	39	AD ₁₅
AD ₁₃	3	38	AD ₁₆ /S ₃
AD ₁₂	4	37	AD ₁₇ /S ₄
AD ₁₁	5	36	AD ₁₈ /S ₅
AD ₁₀	6	35	AD ₁₉ /S ₆
AD ₉	7	34	BHE'/S ₇
AD ₈	8	33	MN/MX'
AD ₇	9	32	RD'
AD ₆	10	31	RQ'/GT'
AD ₅	11	30	RQ'/GT'
AD ₄	12	29	WR'
AD ₃	13	28	M/I/O'
AD ₂	14	27	DT/R'
AD ₁	15	26	PEN'
AD ₀	16	25	ALE
NMI	17	24	INTA'
INTR	18	23	
CLK	19	22	READY
GND	20	21	RESET

Saathi

* Practical Related Questions :

- 1] Name the interrupt of 8086.
- a) NMI - Non-Maskable Interrupt
b) INT - Maskable Interrupt having low priority
- 2] State DMA Controller signals of 8086.
- The direct Memory Access Interface of the 8086 consist of HOLD and HLDA signals.
- 3] RD Signal is active at low (low/high)
- RD Signal is active at low (low/high)

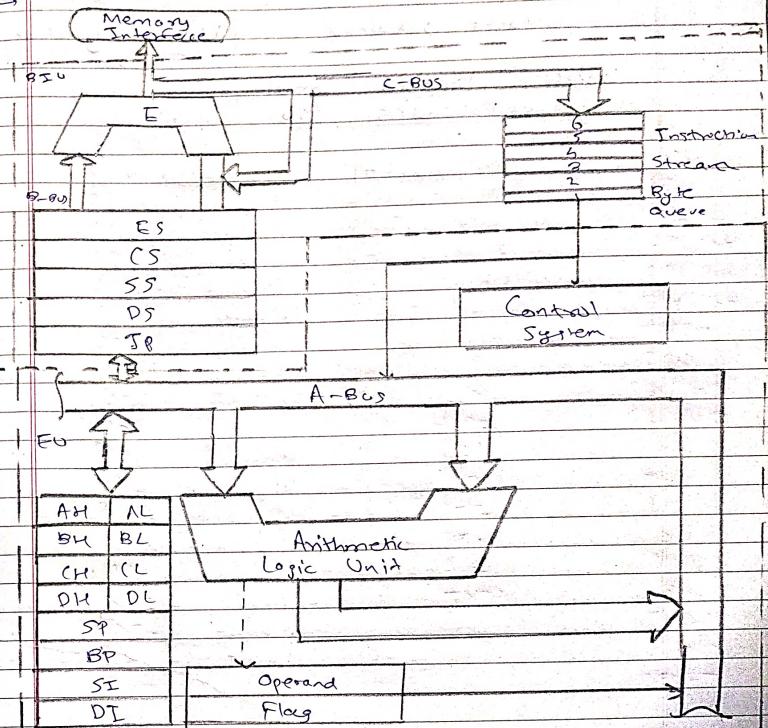
* Exercise

- 1] State the use of status signal in maximum mode.
- i) The 8086 can work in maximum mode by grounding MN/MX.
- ii) In maximum mode, the pins 24 to 31 are defined.
- iii) S_0, S_1, S_2 are status signals.
- iv) These are status signals and they are used by 8288 bus controller to generate the bus timing and control signals.

Date / /

Saathi

- 2] Draw the labelled block diagram of 8086.



2. Use Assembly language Programming Tools and Function

* Observations

- 1] Observe and write content of Register using debugger TD or Debug.

Types	Registers	Flag Register	
General Purpose Register	Ax 0000 Bx 0000 Cx 0000 Dx 0000	Carry Flag Zero Flag Sign Flag Overflow Flag	CF 0 ZF 0 SF 0 OF 0
Index Register	SI 0000 DI 0000	Parity Flag Auxillary Carry flag	PF 0 AF 0
Base Pointer	BP 0000	Interrupt Carry flag	IF 1
Stack Pointer	SP 0000	Direction Flag	DF 0
Segment Register	DS 489D ES 489D SS 48AC CS 48AE		
Instruction Register	IP 0000		

2] Observe and write the contents of memory location in Code Segment using Debugger TD or Debug

Address	Content	Address	Content
CS:0000	CD20	CS:0008	FFADDECB
CS:0001	CD20	CS:0009	FFADDECB
CS:0002	FF9FO00EA	CS:000A	FFADDECB
CS:0003	FF9FO00EA	CS:000B	2000
CS:0004	FF9FO00EA	CS:000C	2000
CS:0005	FF9FO00EA	CS:000D	2000
CS:0006	FF9FO00EA	CS:000E	2000
CS:0007	FFADDECB	CS:000F	0AA20189

3] Observe and write the content of memory location in Data Segment using Debugger TD or Debug,

Address	Content	Address	Content
DS:0001	C020FFAF00EAFFFF	DS:0008	ADDECB2000F000A
DS:0002	C020FF9FO00EAFFFF	DS:0009	A00EC820000F000A
DS:0003	C020FF9FO00EAFFFF	DS:000A	A201490220109201
DS:0004	C020FF9FO00EAFFFF	DS:000B	A201490220109201
DS:0005	C020FF9FO00EAFFFF	DS:000C	A201890220109201
DS:0006	C020FF9FO00EAFFFF	DS:000E	A201890220109201
DS:0007	C020FF9FO00EAFFFF	DS:000F	A201890220109201

Practical Related Questions

1] Write the assembly language tool used in your lab.

Sr.No.	Tools Used	Name of Tool	Version
1	Editor	Notepad	7.2.5
2	Assembler	TASM	5.4
3	Linker	TLINK	5.4
4	Debugger	TD	5.4

2] List the file extensions that are created by the Assembly used.

- - .asm, • .exe, • .hex, • .inc, • .lib, • .obj
 - .lst, • .o, • .mcp, • .mcw

3] List the file extensions that are created by the linker used.

- - .sdi
 - .seg
 - .SEG
 - .set

3. Write an Assembly language program to Perform addition and substraction of two 8 bit and 16 bit number

* Observations

- 1) Observe and write the contents of Register using debugger TD or Debug after the execution of program

Registers		Flag Register	
	AFTER	Before	
AX	4809	0000	Carry Flag CF 0
BX	0002	0000	Zero Flag ZF 0
CX	0009	00000000	Sign Flag SF 0
DX	0000	000000	Overflow Flag OF 0
SI	0000	0000	Parity F PF 0
DI	0000	0000	Auxiliary F AF 1
BP	0000	0000	Interrupt F IF 1
SP	0080	0080	Direction F DF 0
DS	48AD	4893	
ES	489D	4893	
SS	48AC	4893	
CS	48AE	4893	
IP	0087	0100	

- 2) Observe and write the content of memory location in code Segment using Debugger TO or Debug after Execution of program

Address	Contents	Address	Contents
(CS: 0000)	B8AD48	CS: 0008	9A1E0103
(CS: 0001)	B8AD48	CS: 0009	9A1E0100
(CS: 0002)	B8AD48	CS: 000A	A30200
(CS: 0003)	SEDP	CS: 000B	A30200
(CS: 0004)	SEDP	CS: 000C	02C3
(CS: 0005)	A00000	CS: 000D	0000
(CS: 0006)	A00000	CS: 000E	9AC8
(CS: 0007)	A00000	CS: 000F	9AC8

- 3) Observe and write the content of memory location in Data Segment using Debugger.

Address	Contents	Address	Contents
DS: 0000	C020FF9F00EAFFFF	DS: 0008	ADDEE001CS1589
DS: 0001	C020FF9F00EAFFFF	DS: 0009	ADDEE001CS1589
DS: 0002	C020FF9F00EAFFFF	DS: 000A	C51589022010920
DS: 0003	C020FF9F00EAFFFF	DS: 000B	C51589022010920
DS: 0004	C020FF9F00EAFFFF	DS: 000C	C51589022010920
DS: 0005	C020FF9F00EAFFFF	DS: 000D	C51589022010920
DS: 0006	C020FF9F00EAFFFF	DS: 000E	C51589022010920
DS: 0007	C020FF9F00EAFFFF	DS: 000F	C51589022010920

* Program Code with Comments

1) ; addition
 data Segment
 a db 07h
 b db 02h
 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 cl, al
    mov c, al
```

Code ends
 end start

2) ; Subtraction
 data Segment
 a db 07h
 b db 02h
 c dw ?
 data ends

Date _____

Saathi

Code segments
assume CS:code, DS:Data

```

start:
    mov ax, data
    mov ds, ax
    mov al, a
    mov bl, b
    add al, bl
    sub al, bl
    mov cl, al
    mov cx, ax
    code ends
end start

```

Results

- 1] $7+2=9$
- 2] $7-2=5$

Practical Related Questions

- 1) Write the command line used to create ALP for addition and subtraction.
- The command line used to create ALP for addition and subtraction are:-

Add AL, BL
Sub AL, BL

- Date _____
- Saathi
- 2] Write the command line used to generate .exe file of assembly language program for addition and subtraction.
 - Data Segment and Code Segment are used to generate .exe file of Assembly language Programming.
 - 3] Write assembler directives used in program
→ END, DB,
Data directives :- DB, DW
ASSUME

Exercise

- 1] Write the content of AL register and status of Flags after execution of following code
MOV AL, 99
ADD AL, 01
→ AL Register - 0063
ADD - 0064
Status of Flag - No change

2) Write an ALP for addition of two 32 bit numbers

data segment

abc dd 12345678h

def dd 9ABCDEF0h

ghi dw ?

data ends

Code segment

assume cs:code, ds:data

start:

mov ax, data

mov ds, ax

mov di, 00h

mov ax, word ptr abc+2

mov bx, word ptr def

add ax, bx

mov word ptr ghi, ax

mov ax, word ptr abc+2

mov bx, word ptr def+2

add ax, bx

mov word ptr ghi+2, ax

jnc more

inc di

more: mov byte ptr ghi+4, dl

int 3

Code ends

end start

Saath

Saathie

Date: 1/1/2023

4. Write an ALP to multiply two given 8 and 16 bit unsigned and signed number

* Observations

Observe and write contents of Register using using debug after the execution of program

	Registers		Flag Register
	After	Before	
AX	0008	0000	Carry Flag CF 0
BX	0004	0000	Zero Flag ZF 0
CX	0000	0000	Sign Flag SF 0
DX	0000	0000	Overflow flag OF 0
SI	0000	0000	Parity Flag PF 0
DI	0000	0000	Auxiliary Carry AF 0
BP	0000	0000	Interrupt flag IF 1
SP	0000	0000	Direction Flag DF 0
DS	48AD	48AD	
ES	489D	48AD	
SS	48AC	48AC	
CS	48AE	48AE	
IP	0010	0000	

Table 2

Address	Contents	Address	Contents
DS: 0000	0020FF9E00EAFFFF	DS: 0008	A01FF001C512E0
DS: 0001	C020FF9E00EAFFFF	DS: 0009	A0DE001C515B0
DS: 0002	C020FF9E00EAFFFF	DS: 000A	C51589022010A90
DS: 0003	C00EFF9E00EAFFFF	DS: 000B	C51589022010A90
DS: 0004	C020EFF9E00EAFFFF	DS: 000C	C51589022010A90
DS: 0005	C020EFF9E00EAFFFF	DS: 000D	C51589022010A90
DS: 0006	C020EFF9E00EAFFFF	DS: 000E	C51589022010A90
DS: 0007	C020FFF9E00EAFFFF	DS: 000F	C51589022010A90

Program Code with Comment

```

data segment
    a dw 0AH
    b dw 0BH
    c dd ?
data ends

Code segment
assume ds: data, cs: code
start:
    mov ax, data
    mov ds, ax
    mov ax, a
    mov bx, b
    imul bx
code ends
end start

```

Saathi

Date

Saathi

★ Result:

$$2 \times 4 = 8$$

★ Practical Related Questions

- 1] Write the names of result registers of multiplication of 8/16-bit unsigned and signed number.

→ Register Ax was used to show result

- 2] Which instructions you have used to extend the sign of 8-bit negative number for 8-bit * 16-bit multiplication.

→ IMUL instruction was used.

★ Exercise

- 1] Write the content of Ax register after execution of following code.

```

MOV BL, -1
MOV RL, 1
IMUL BL

```

$$AX = 4801$$

- 2] State the flag affected by IMUL instruction

→ Sign Flag and Parity Flag were affected

- Date / /
- Saathi
- i) State the difference between MUL and IMUL
 - i) The MUL and IMUL are multiplication instruction
 - i) MUL instruction is used for multiplying unsigned numbers
 - iii) On the other hand, IMUL multiplies signed numbers

Assessment Scheme

Mark obtained	Dated Signature of Teacher	
Process Related (25)	Product Related (15)	Total (50)

Date / /

Saathi

5. Write an ALP to perform block transfer operation

Observations

- i) Memory location before transfer

Source Memory Block		Destination Memory Block	
Address	Content	Address	Contents
DS: 0000	56H	DS: 0005	15H
DS: 0001	7BH	DS: 0006	49H
DS: 0002	62H	DS: 0007	F7H
DS: 0003	23H	DS: 0008	C9H
DS: 0004	AAH	DS: 0009	55H

- ii) Memory location after transfer

Source Memory Block		Destination Memory Block	
Address	Content	Address	Content
DS: 0000	56H	DS: 0005	56H
DS: 0001	7BH	DS: 0006	7BH
DS: 0002	62H	DS: 0007	62H
DS: 0003	23H	DS: 0008	23H
DS: 0004	AAH	DS: 0009	AAH

* Program with Comment

Data Segment
 String 1 DB 01H, 02H, 03H, 04H, 05H
 String 2 DB 4 DUP(0)
 Data Ends

Code Segment
 Assume CS:Code, DS:Data
 Start: MOV AX, Data
 MOV DS, AX
 MOV CS, AX
 LEA SI, String1
 LEA DI, String2
 MOV CX, 05H
 CLD
 REP MOVS
 INT 3
 Code ENDS
 END Start

* Result.

Block transfer Successful.

* Practical Related Questions

- 1) Write the instructions you have used to initialize memory pointer for source and destination block of data.
 → MOVS| MOVB| MOVSW are the instructions used to initialize memory.
- 2) State the name of Register which is used as Counter.
 → The register CX was used as Counter

* Exercise

- 1) Write an ALP to perform block transfer of overlapping block.
 →
- data segment
 b1K1 dd 10 dup (?)
 b1K2 db 10 dup (?)
 data ends

Code Segment

Start: Assume CS:Code, DS:Data
 mov ax, data
 mov ds, ax
 lea si, b1K1
 lea di, b1K2
 mov cl, 0ah
 mov bx, 05h

Date / /

```

mov di, bx
up: mov al, [si]
mov [di], al
inc di
inc si
dec cl
jnz up
mov ah, 4ch
int 21h
code ends
end start

```

Saathi

Assessment Scheme

Mark Obtained _____ Dated _____
Signature of Teacher _____

Process	Product	Total (50)
Related (35)	Related (15)	
		(17) good 16.5 bad
		(18) with 16.5 bad

Date / /

6. Write an ALP to compare
two strings

Saathi

Observations :

Observe and write the contents memory location
of Source and destination string

Source Memory Block		Destination Memory Block	
Address	Contents	Address	Contents
DS:0000	56 H	DS:0008	C9H
DS:0001	7B H	DS:0009	55H
DS:0002	62 H	DS:000A	A4 H
DS:0003	23 H	DS:000B	43 H
DS:0004	A4H	DS:000C	69 H
DS:0005	15 H	DS:000D	77 H
DS:0006	42 H	DS:000E	C4 H
DS:0007	F7 H	DS:000F	D6 H

Date _____

Saathi

* Program Code :

Data Segment

```
STR1 db 'GLSICT\$, \$1$V'
STRLEN1 db $-STR1
STR2 db 'GLSICT\$, \$1$V'
STRLEN2 db $-STR2
String db 'Strings are Equal\$, \$1$V'
String1 db 'Strings are Unequal\$, \$1$V'
Data ends
```

Code Segment

Assume CS:code, DS:DATA

Begin:

```
MOV AX, DATA
MOV DS, AX
MOV ES, AX
LEA SI, STR1
LEA DI, STR2
MOV CX, 6
MOV AL, STRLEN1
MOV BL, STRLEN2
CMP AL, BL
JNE Not_Equal
JPEP CMPSB
JNE Not_Equal
JMP Equal
Not_Equal:
    MOV AH, 0AH
    LEA DX, String
    INT 21H
```

Date _____

Saathi

jmp Exit

Equal:

```
MOV AH, 0AH
LEA DX, String
INT 21H
```

Exit:

```
MOV AX, 4C00H
INT 21H
```

Code ends

End Begin

* Result :

Comparison of two strings done

* Practical Related Questions :-

1] Write the instructions you have used to initialize memory pointer for source and destination string.

→ STR1 db 'GLSICT\\$, \\$1\$V'

2] Write the string instruction used for string comparison in your program.

→ CMPS/ COMPB / CMPSB were instructions used.

(Saathi)

* Exercise

1) Write the registers used as memory pointer for source and destination while using string compare instruction CMPS.
 → ES: DI and DS: SI pair of registers are used.

2) Write the flag which are used to know whether string are equal or unequal
 → Zero flag and Carry Flag are used.

3) State the uses of REP prefix instruction
 → REP is used for repeating an instruction count number of times.

* Assessment Scheme

Mark Obtained	Dated Signature of Teacher
---------------	-------------------------------

Process	Product	Total (50)
Related (35)	Related (15)	total 50

(Saathi)

Date: / /

7. Write an ALP to divide two 8 bit and two 16 bit unsigned and signed numbers.

* Observations

Registers	Flag Register
AX	Carry Flag CF 0
BX	Zero Flag ZF 0
CX	Sign Flag SF 1
DX	Overflow Flag OF 0
SI	Parity Flag PF 1
DI	Auxiliary Carry Flag AF 0
BP	Interrupt Flag IF 1
SP	Direction Flag DF 0
DS	
ES	
SS	
CS	
IP	

2) Observe and write contents of memory location in Data Segment.

Address	Contents	Address	Content
DS:0000	C00FF9F0UFAFF	DS:0008	ADDEE001CS15
DS:0001	C020FF9F0UCA	DS:0009	APREE001CS15
DS:0002	C020FF9F0UCA	DS:000A	CS158902201920
DS:0003	D020FF9F0UFA	DS:000B	CS158902201920
DS:0004	C020FF9F0UFA	DS:000C	CS158902201920
DS:0005	C020FF9F0UFA	DS:000D	CS158902201920
DS:0006	C020FF9F0UFA	DS:000E	CS158902201920
DS:0007	C020FF9F0UFA	DS:000F	CS158902201920

→ Program Code with Comments

```
data segment
a dw 4444h
b dw 0002h
c dw ?
data ends
```

Code Segment

```
assume ds: data, cs: code
start:
    mov ax, data
    mov bx, b
    div bx
    mov c, ax
    int 3
code ends
end start
```

Saathi

Date _____

* Result

$$10 \mid 2 = 5$$

* Practical Related Questions

1) Write the result of division of signed number you have taken in program.
→ 5

2) State the difference DIV and IDIV instruction
→ The DIV instruction divides unsigned numbers and IDIV divides signed numbers. Both return a quotient and a remainder. The dividend is the number to be divided and the divisor is number to divide by. The quotient is result.

* Exercise

2) Write and ALP to divide 16-bit signed number by 16-bit signed number.
→

```
data segment
a dw 4444h
b dw 0002h
c dw ?
data ends
```

```
Code Segment
assume ds: data, cs: code
```

Page No.

Page No.

Start:
 mov ax,data
 mov ds,ax
 mov ax,a
 mov bx,b
 dN bx
 mov c,ax
 int 3
 Code ends
 end start

Assessment Scheme

Mark Obtained	Parted Signature of Teacher	
Process	Product	Total (so)
Related (25)	Related (15)	

Saathi

Date _____ / _____ / _____

8. Write an ALP to perform arithmetic operation on BCD Number

& Observations

1] Observe and write the content of Register Using Debug after the execution of program

Registers		Flag Register	
	After	Before	
AX	4C98	0000	Carry Flag CF 1
BX	0099	0000	Zero Flag ZF 0
CX	0000	0000	Sign Flag SF 1
DX	0000	0000	Overflow Flag OF 1
SI	000000	0000	Parity Flag PF 0
DI	0000	0000	Auxiliary Carry Flag AF 1
BP	0000	0000	Interrupt Flag IF 1
SP	0400	0000	Direction Flag DF 0
DS	48A0	48A3	
ES	48A0	48A3	
SS	48AF	48A3	
CS	48A0	48A3	
IP	0009	0000	

Date: / / *Saathi*
 7) Observe and write the content of memory location in Data Segment after the execution of Program.

Address	Contents	Address	Contents
DS:0000	C020FFAFOUFAFF	DS:0004	ADPEE001CS15A
DS:0001	C040FFAFOUFBFFF	DS:0009	ADPEE001CS15A
DS:0002	C020FFAFOUFEFFFF	DS:000A	CS158902200920
DS:0003	C020FFAFOUFAFFF	DS:000B	CS1589022010920
DS:0004	C020FFAFOUFAFFF	DS:000C	CS1589022010920
DS:0005	C020FFAFOUFAFFF	DS:000D	CS1589022010920
DS:0006	C020FFAFOUFAFFF	DS:000E	CS1589022010920
DS:0007	C020FFAFOUFAFFF	DS:000F	CS1589022010920

* Program Code with Comments

```
; BCD Addition
; Model Small
; Stack
; Code
MOV AL, 9AH
MOV BL, 9AH
ADD AL, BL
DAA
```

```
MOV AH, 4CH
INT 21H
END
```

Saathi

* Result:

→ Result in BCD Addition was done

* Practical Related Questions

1) Write the flags used for BCD arithmetic operations.

→ Auxiliary Carry Flag (AC) is used for BCD arithmetic operations

2) Write the instruction that converts the result of addition and subtraction in Unpacked decimal digit.

→ ADD is the instruction that converts the result of addition and subtraction in unpacked decimal digit.

* Exercise

1) Write the output of DAA instruction in AL register of following code after execution and also of status of CF and AF.

```
MOV AL, 9AH
MOV BL, 01H
ADD AL, BL
DAA
```

→ AL = 9A

CF = 1

AF = 1

Date: _____

2) Write an output of DAS instruction in AL register of following code after the execution. And also status of CF and AF.

MOV AL, 03H

MOV BL, 01H

SUB AL, BL

DAA

→ AL = FC CF = 1 AF = 1

CF = 1

AF = 1

* Assessment Scheme

Mark Obtained	Dated Signature of Teacher	
.....	
Process	Product	Total (Sc)
Related (35)	Related (35)
.....

Final Submission Date: 30/04/2017 On 4x6110

and later with help of counselor & the respective

classmate, I have done the assignment

and submitted it on time.

I am very happy to submit my assignment.

Thank you for your support.

With regards,

SAATHI

Saathi

Date: _____

9. Implement Loop to find Sum of Series of BCD numbers

Saathi

* Observations

1) Observe and write the content of Register using Debug after execution of program.

	Registers	Flag Register	
	After	Before	
AX	000A	0000	Carry Flag CF
BX	0002	0000	Zero Flag ZF
CX	0000	0000	Sign Flag SF
DX	0000	10000	Overflow Flag OF
SI	0000	0000	Parity Flag PF
DI	0000	0000	Auxiliary Carry Flag AF
BP	0000	0000	Interrupt Flag IF
SP	0000	0000	Direction Flag DF
DS	4893	4893	
ES	4893	4893	
SS	4893	4893	
CS	4893	4893	
IP	0005	0000	

2) Observe and write the content of memory location in Data Segment after the execution of program.

Address	Content	Address	Content
DS:0000	C020FF00FAFFFF	DS:0008	A00E001CS15A
DS:0001	C020FF00FAFFFF	DS:0009	A00E001CS15A
DS:0002	C020FF00FAFFFF	DS:000A	CS1589 0220109201
DS:0003	C020FF00FAFFFF	DS:000B	CS1589 0220109201
DS:0004	C020FF00FAFFFF	DS:000C	CS1589 0220109201
DS:0005	C020FF00FAFFFF	DS:000D	CS1589 0220109201
DS:0006	C020FF00FAFFFF	DS:000E	CS1589 0220109201
DS:0007	C020FF00FAFFFF	DS:000F	CS1589 0220109201

* Program Code

```

• Model Small
• data
Res DB ?
CR1 DB 0AH
• code
Start: MOV AX, @data
MOV DS, AX
LEA SI, RE1
MOV CL, CNT
MOV AX, 00H
MOV BX, 01H
L1: ADD AX, BX
MOV [SI], AX

```

(Saathi)

Date _____ / _____ / _____

(Saathi)

MOV AX, BX

MOV BX, [SI]

INC SI

LOOP L1

INT 3H

END START

* Result

Looping was used in program

* Practical Related Questions

1) Note down the registers used as memory pointer and counter in a program?

→ AX and BX register were used as memory pointer and CL register was used as Counter

2) State the use of INC instruction in program.

→ The INC instruction adds one to the destination operand, while preserving the state of the Carry Flag.

* Exercise

1) What is the condition to terminate loop formed using loop statement.

→ If the result created in register is zero, the loop is terminated and execution is continued.

Q) Write applications where loop instruction can be used.
 → The loop instruction is mainly used to simulate the different loops in HLL.
 The loop instruction uses the CX register to indicate the loop Count. The ECX register can be used as a loop counter as it is 32 bit.

Q) Which register is used as a Counter to store count for a loop instruction?
 → CL Register is used as Counter.

Assessment Scheme

Marks obtained			Dated Signature of Teacher
Process	Product	Total (so)	
Related (5)	Related (15)		

10. Write an ALP to

Find Smallest number from an array of n numbers

A. Observations

1) Observe and write the contents of memory location and AL register after execution of program.

Addresses	Original Content	Loop 1	Loop 2	Loop 3	Loop 4	Loop 5
DS:0000	12					
DS:0001	07					
DS:0002	25	AL = 12	AL = 07	AL = 25	AL = 18	AL = 02
DS:0003	18					
DS:0004	02					

2) Observe and write the content of memory location and AL register after execution of program.

Addresses	Original Content	Loop 1	Loop 2	Loop 3	Loop 4	Loop 5
DS:0000	12					
DS:0001	07					
DS:0002	25	AL = 02	AL = 18	AL = 25	AL = 07	AL = 12
DS:0003	18					
DS:0004	02					

* Program Code

```
model small
data
array db 12h, 4h, 6h, 7h, 8h
```

```
SI db 0
```

* Code

```
mov ax, @data
mov ds, ax
mov cx, 5
mov si, offset array
mov al, [si]
```

```
dec cx
```

```
up:
inc si
cmp al, [si]
jc next
```

```
mov al, [si]
```

next:

loop up

```
mov si, al
ends
end
```

Saathi

Date _____

* Result

12, 4, 6, 7, 8
4 is small

* Practical Related Questions

1] Which Flags are affected by CMP Instruction
→ The Carry Flag, Zero Flag, Sign Flag, Overflow Flag, Parity Flag, Auxiliary Flag are affected by CMP instruction.

2] Which instruction are used to make decision to find smallest number in the program?

→ The CMP decision is used to make the decision to find smallest number in program.

* Exercise

1] Show the Flag status after comparison of following operands

N1	N2	CMP N1, N2			N1	N2	CMP N2, N1		
		CF	ZF	SF			CF	ZF	SF
25	45	1	0	1	75	35	0	0	1
75	45	1	0	0	23	87	1	0	0
734	734	1	1	1	100	100	1	1	1

Q) Write Syntax of CMP instruction with suitable example.

Syntax: CMP destination, Source

Example: CMP DX, 00

Q) Which condition Jump instruction are used to find largest and smallest number?
→ JC and JNC instruction are used to find largest and smallest numbers.

Assessment Scheme

Marks Obtained	Dated Signature of Teacher
----------------	----------------------------

Prices Related (35)	Product Related (15)	Total (50)

11. Write an assembly

language program to arrange number in ascending / descending order.

* Observations

1] Observe and write the contents of memory location after the execution.

Address	Original Content	Pass 1	Pass 2	Pass 3	Pass 4	Pass 5
DS:0000	12	00	00	12	00	00
DS:0001	07	00	07	00	00	00
DS:0002	25	00	00	00	00	25
DS:0003	19	00	00	00	19	00
DS:0004	02	02	07	12	18	25

2] observe and write the contents of memory location after execution

Address	Original Content	Pass 1	Pass 2	Pass 3	Pass 4	Pass 5
DS:0000	12	00	00	12	00	00
DS:0001	07	00	07	00	00	00
DS:0002	25	00	00	00	00	25
DS:0003	18	00	00	00	18	00
DS:0004	02	02	07	12	19	25

* Program Code

```

ASSUME CS:Code, DS:Data
Code Segment
    mov ax, data
    mov ds, ax
    mov dx, count - 1
    Back: mov cx, dx
    mov si, offset list
    Agata: mov ax, [si]
    Cmp Ax, [si + 2]
    Jf go
    XCHG AX, [SI + 2]
    XCHG AX, [SI]
    Go: INC SI
    INC SI
    Loop AGAIN
    DEC DX
    JNZ BACK
    HLT
CODE ENDS
DATA SEGMENT
    list DW 05H, 04H, 01H, 03H, 02H
    COUNT EQU 05H
DATA ENDS
END

```

Saathi

Date / /

Saathi

* Result

List CS: 000H = 05H, 04H, 01H, 03H, 02H
List DS: 000H = 01H, 02H, 03H, 04H, 05H

* Practical Related Questions

- 1] What is the use of XCHG instruction in your program?
→ The XCHG instruction used to exchange content of HL Register pair with DE register pair. This instruction uses implied addressing mode.
- 2] Which Sorting algorithm is used in your program?
→ Bubble Sort algorithm is used in program.

* Exercise

- 1] If numbers in an array are 07H, 02H, 04H, 10H, 06H, write the array contents in each pass while arranging numbers in ascending order.
→ Pass I: 02H, 07H, 04H, 10H, 06H
Pass II: 02H, 07H, 09H, 06H, 10H
Pass III: 02H, 07H, 06H, 04H, 10H
Pass IV: 02H, 06H, 07H, 09H, 10H

Ascending order: 02H, 06H, 07H, 09H, 10H

Q) If numbers in an array are 07H, 02H, 09H,
10H, 08H write the array contents in
each pass while arranging in descending order.

Pass I: 07H, 02H, 10H, 09H, 08H
Pass II: 02H, 10H, 07H, 09H, 08H
Pass III: 10H, 07H, 02H, 09H, 08H
Pass IV: 10H, 02H, 09H, 07H, 08H
Pass V: 10H, 08H, 07H, 09H, 02H

Descending order: 10H, 09H, 07H, 08H, 02H

* Assessment Scheme

	Mark obtained	Dated Signature of Teacher
Prizes	Product	Total (50)
Related (25)	Related (15)	Related (10)
Related (10)	Related (10)	Related (10)

Saathi

Saathi

Q) Write an assembly language program to find length of string, arrange string in reverse order and concatenate strings.

* Observations:

1) Reverse of a String

	Example 1	Example 2
Input string taken	MSBTE	TEST
Reverse string	E-TBSM	TSET

2) Length of String

	Example 1	Example 2
Input string taken	Microprocessor	Test
Length of String	14	4

3) String Concatenation

	Example 1	Example 2
Input String 1 taken	Microprocessor	Test subject
Input String 2 taken	Programming	Subject
Output string	Microprocessor programming	Test subject

Saathi

* Program code :

; Program to find length of string
• Model Small
• Stack 100H
• Data
STR1 DB "MSBTE\$"
LEN DB 0
• Code
MOV AX, @DATA ; Initialising Data Segment
MOV DS, AX
MOV SI, offset STR1

UP:
MOV AL, [SI]
CMP AL, 10H
JZ DN
INC LEN
INC SI
Loop UP

DN:
; Printing length
MOV DL, LEN
ADD DL, 48
MOV AH, 2
INT 21H
MOV AH, 4CH; Service Routine for exit
INT 21H
END

Date _____ / _____ / _____

Saathi

* Result :

5

* Practical Related Questions :

- 1) State the registers that are used as memory pointer in String reverse and Concatenation program.
→ Register AX and SI are used
- 2) State the role of direction flag while using string instruction.
→ The direction flag is the flag that controls the left to right or right to left direction of string processing, stored in the flags register on all x86 compatible CPUs.

* Exercise

- 1) Write an ALP to find the string is palindrome or not.
→
 - MODEL STACK
 - STACK 100H
 - DATA
 - STRING DB 'abba','\$'
 - STRING1 DB 'String is Palindrome!', '\$'
 - STRING2 DB 'String is not Palindrome!', '\$'

Date / /

Code

```

MAIN PROC FAR
MOV AX, @DATA
MOV DS, AX
CALL PALINDROME
MOV AH, 4CH
INT 21H
MAJN 8000
Palindrome proc
Loop 2:
CMP SF, DI
JC OUTPUT1
MOV AX, [SI]
MOV BX, [DI]
CMP AL, BL
JNE OUTPUT2
OUTPUT1:
LAA DX, STRING1
MOV AH, 0AH
INT 21H
RET

```

Palindrome ENDP

END MAIN

Mark obtained	Paired Signature	
Process	Product	Total (50)
Related (35)	Related (15)	

Page No.

Date / /

(Saathi)

13. Write an ALP to count odd or even number in array

Observations:

I] Observe and write the content of Register using Td.

	Register	Flag Register
	AF	CF
AX	0192	Carry Flag
BX	000B	Zero Flag
CX	F70F	Sign Flag
DX	698D	Overflow Flag
SI	F712	Parity Flag
DI	F713	Auxiliary Carry Flag
BP	0100	Interrupt Flag
SP	0106	Direction Flag
DS	2110	CF
ES	012D	ZF
SI	0192	SF
CS	0000	OF
IP	0100	PF

Page No.

g) Observe and write the content of memory locations in Data Segment after the execution of program.

Address	Contents	Address	Content
DS:0000P	CD20FF9E01FAFF	DS:0008	ADD E DX, 01CS15
DS:0001	DS:0009	DS:0009	SJ
DS:0002	DS:000A CS1587002010920	DS:000B	"
DS:0003	DS:000C	DS:000C	"
DS:0004	DS:000D	DS:000D	"
DS:0005	DS:000E	DS:000E	"
DS:0006	DS:000F	DS:000F	"
DS:0007			

* Program Code

Data Segment

A DW 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Data ends

Code Segment

Assume DS:Data CS:Code

Start:

MOV AX, Data

MOV DS, AX

LEA SI, AX

MOV DX, 0000

MOV BL, 02

MOV CL, 10

L1: MOV AX, WORD PTR[SI]

DIV BL
CMP AH, 00

JNZ L2

INC DH

JMP L3

L2: INC DL

L3:

ADD SI, 2

DEC CL

CMP CL, 00

JNZ L1

MOU AH, DL

INT 21H

Code Ends

* Output:

5 - Even

5 - odd

* Practical Related Questions:

1) Write the flag used to check whether the number is odd or even.

→ The zero flag is used to check whether the number is odd or even.

2) Which bit of 8/16-bit number is used to decide if number is odd or even?

→ 8-bit number is used.

(Saathi)

- Exercise
- 1) Write an ALP to count odd as well as even number in array of 10 numbers.

→ Data Segment

```

N DB 78
RES DB ?
DATA ENDS
CODE SEGMENT
ASSUME CS:CODE, DS:DATA
START: MOV AX, DATA
        MOV DS, AX
        MOV AL, 0DH
        MOV AH, 0AH
        MOV BL, 02H
        DJNZ BC
        CMP AL, 00H
        JE L1
        MOV RES, 00H
        JMP L2
L1: MOV RES, 00FFH
L2: INT 3
CODE ENDS
END START
    
```

(Saathi)

- Date / /
14. Write an ALP to count Positive and/or Negative number in Array

→ Observations :

- 1) Observe and write content of Register using TD after execution of program.

	Register		Flag Register	
	After	Before		
Ax	0003	0000	Carry Flag	CF 1
BX	000A	0000	Zero Flag	ZF 0
CX	00B4	0000	Sign Flag	SF 0
DX	0004	0000	Overflow Flag	OF 1
SI	0000	0004	Parity Flag	PF 1
DI	0000	0000	Auxiliary carry Flag	AF 0
BP	0000	0000	Interrupt Flag	IF 1
SP	0000	0000	Direction Flag	DF 1
DS	489D	489D		
ES	489D	489D		
SS	48AC	48AC		
CS	48AF	48AE		
IP	0018	0000		

Mark Obtained

Dated

Process Related	Product Related	Total	Signature of Teacher

Page No.

Page No.

2) Observe and write the Content of memory location in Data Segment after the execution of program.

Address	Contents	Address	Contents
DS:0000	C020FF & F04CAFEE	DS:0008	ADDEE C001C515
DS:0001		DS:0009	
DS:0002		DS:000A	C515 89 00201091
DS:0003		DS:000B	
DS:0004		DS:000C	0000 0000
DS:0005		DS:000D	
DS:0006		DS:000E	
DS:0007		DS:000F	0000 0000

* Program Code

```

data segment
num db -3,1,-5,6,7,9,11
p_cnt db 0h
n_cnt db 0h
data ends

```

Code segment

```

assume ds: data, cs: code
start:
    mov ax, data
    mov ds, ax
    lea si, num

```

(Saathi)

Date _____

main:

```
    cmp nom[si], 0h
```

```
    jg pos
```

```
    inc si
```

```
    add n_cnt, 0h
```

```
    cmp num[si], 11#11
```

```
    je exit
```

```
    jmp main
```

```
pos:
```

```
    add p_cnt, 0h
```

```
    inc si
```

```
    cmp nom[si], 11#11
```

```
    je exit
```

```
    jmp main
```

exit:

```
    mov bl, p_cnt
```

```
    mov cl, n_cnt
```

```
    mov ax, 6C00h
```

```
    int 16h
```

Code ends

end start

* Result:

Positive Number - 1, 6, 9

Negative Number - -3, -5, -7

Saathi

Date _____

Practical Related Questions

- 1) Write a program which is used to check whether the number is positive or negative.
- Zero Flag was used to check whether the number is positive or negative.
- 2) Which bit of 8-bit/16-bit number is used to decide if the number is positive or negative?
- 8-bit Counter is used to decide.

Exercise

- 1) Write an ALP to count Positive and negative numbers in array of 10 numbers.

→ Model Small
→ Data

```
CSEG
    DB 0DH
    DB 0AH
    LIST DW 0579H, 0A500H, 0C009H, 0159H, 0B9CH
    COUNT EQU 0SH
```

→ Code

```
MOV AX, @DATA
MOV DS, AX
XOR BX, BX
XOR DX, DX
MOV CL, COUNT
MOV SI, OFFSET LIST
```

Date _____

Again: MOV AX, BX

SHL AX, 01

JC negat

JNC BX

JMP NEXT

NEGAT: INC BX

NEXT:

ADD SE, 02

DEC CL

JNZ Again

Mov AH, NC

INT 21H

END

Assessment Scheme:

Marks Obtained		Dated Signature of Teacher
Practical Related	Product Related	Total

Date / /

Saathni

15. Write an ALP to count 10's and '1's in a given number.

Observations

- 1] Observe and write the contents of Register using debugger TD.

Register		Flag Register	
	After	Before	
Ax	9C39	0000	Carry Flag CF 1
BX	0008	0000	Zero Flag ZF 1
CX	0000	0000	Sign Flag SF 0
DX	0008	0000	Overflow Flag OF 0
SI	0000	0000	Parity Flag PF 1
DI	0000	0000	Auxiliary Carry Flag AF 0
BP	0000	0000	Interrupt Flag IF 1
SP	0000	0000	Direction Flag DF 0
DS	48A0	4890	
ES	4890	4890	
SS	48AC	48AC	
CS	48AF	48AF	
IP	0018	0000	

Saathi

Q) Observe and write the contents of Register using Debugger TD or Debug after the execution of program.

Number of 8bit/10 bit in Hexadecimal	ND. of 1's	No. of 0's
AA	4	4
55	2	6
33	4	4
99	4	4
FFFF	8	8
AA55	8	8
FOFO	8	8
9898	5	5

* Program Code:

```

Data Segment
Number DW 9C94
Zero DW 0100H (?)
One DW 0100CH (?)
DATA ENDS

```

```

CODE SEGMENT
ASSUME CS: CODE, DS: DATA
START:

```

```

    MOV AX, DATA
    MOV DS, AX
    MOV AX, NUMBER

```

Date / /

```

MOV BX, 00H
MOV CX, 00H
MOV DX, 00H

```

LABEL1:

```

ROL AX, 1
JC ONE
INC BX
JMP NXT

```

ONE:

```

INC DX

```

NEXT:

```

DEC CX
JNZ LABEL1
MOV ZERO BX
MOV ONE DX
MOV AH, 2CH
INT 21H
CODE ENDS
END START

```

* Result:

Count of 1's - 8
Count of 0's - 9

* Practical Related Questions

- 1) Write the flag used to count '1's and '0's.
 → Carry Flag was used to Count '1's and '0's

Process Related	Product Related	Total	214%
Date			(Saathi)

- Q) Write the Instructions used in your program to rotate and check number or 101
17.

→ JMP - Jump
JNZ - Jump NZE zero
ROL - Rotate Left

* Exercise

- 1) Modify your program to Count Number of 101 in AL register.

→ Data Segment

Num DB 5G4

Zeroes DB?

DATA ends

CODE Segment

Assume CS: DATA, DS: DATA

Start:

MOV DS, AX

MOV AL, NUM

MOV BL, 0AH

MOV CX, 008H

UP: ROL AL, 01H

INC BL

JMP Next:

Next: MOV ZEROS, BL

JNZ UP

COD END.S

END. Start

Date _____ / _____ / _____ (Saathi)
16. Write an assembly language program using procedure

* Observations

- 1) Observe and write Content of Registers CSIG TO after execution of program

Registers		Flag Register	
	After	Before	
AX	0004	0000	Carry Flag CF 0
BX	0006	0000	Zero Flag ZF 1
CX	0000	0000	Sign Flag SF 1
DX	0005	0000	Overflow Flag OF 0
SI	0000	0000	Parity Flag PF 1
DI	0000	0000	Auxiliary Carry Flag AF 1
BP	0000	0000	Interrupt Flag IF 1
SP	0000	0000	Direction Flag DF 1
DS	48AD	489D	
ES	489D	489D	
SS	48AC	48AC	
CS	48AE	48AE	
IP	016	0100	

Q) Observe and write the contents of memory location in Data Segment AFTER execution

Address	Contents	Address	Contents
DS:0000	C020FFEA0000FFFF	DS:0008	ADDEE001C515
DS:0001		DS:0009	
DS:0002		DS:000A	CS15 89 02 01 92 0
DS:0003		DS:000B	
DS:0004		DS:000C	
DS:0005		DS:000D	
DS:0006		DS:000E	
DS:0007		DS:000F	

* Program Code:

```
Section .text
global _start
_start:
    mov ecx, 14
    sub ecx, 0
    mov edx, '5'
    sub edx, '0'
    call sum
    mov [ecx], eax
    mov ecx, msg
    mov eax, len
    mov ebx, 1
    mov eax, 4
```

Saathi

Date: / /

Saathi

int 0x84

```
    mov ecx, 0x0
    mov edx, 1
    mov ebx, 1
    mov eax, 4
    int 0x80
    mov eax, 1
    int 0x80
sum:
    mov eax, 0x0
    add eax, edx
    add eax, '0'
    ret
```

Section .data

```
msg db "The sum is:", 0xA, 0xD
len equ $ - msg
```

```
Segment .bs
seg real1
```

* Result:

The sum is 9

- Date / /
- Saathi
- * Practical Related Question
- 1) Which procedure have been used in your program (near or far)?
→ Near procedure has been used in program
 - 2) Write the contents of Instruction pointer IP before and after the execution of CALL instruction:
before : 0100 → after : offset 0101
 - 3) What are advantages of using procedure?
→ Reusability of code
Less usage of memory
Development becomes easier
Reduced time
Debugging became easy

* Exercise

- 1) Write an ALP to find smallest number from the array of 10 numbers using procedure.
-
- Model small
 - Stack 10ch
 - data
 - array db 7, 3, 6, 9, 5
 - Code

Date / /

Saathi

```

main proc
    mov ax, 0 data
    mov ds, ax
    mov si, offset array
    mov cx, 5
    mov bl, [si]
    loop X:
        cmpb [SI], $1
        jle update
        inc si
    loop loopx
    print "Smallest value"
    addl bl, 48
    mov dl, bl
    mov ah, 02h
    int 21h
    update:
        mov bl, [SI]
        jmp return
    main endp
End main

```

Marks Obtained		Dated Signature
Process Related	Product Related	Total

Date / /

11. Write an Assembly language program using macros

* Observation :

1) Register

	Register	Flag Register			
	After	Before			
AX	000E	00 00	Carry flag	CF	1
BX	0000	0000	Zero flag	ZF	1
CX	0000	0000	Sign flag	SF	0
DX	0000	00 00	Coverflow flag	OF	0
SI	0000	0000	Parity flag	PF	1
DI	0000	0000	Auxiliary carry flag	AF	0
BP	0000	0000	Interrupt flag	IF	1
SP	0000	0000	Direction flag	DF	0
DS	48A0	48A0			
ES	489D	489D			
SS	48AC	48AC			
CS	4846	48AE			
IP	000C	0000			

2)

Address	Content	Address	Content
DS:0000	CD20FF9FOUEAFFFF	DS:0008	ANREFE001C515
DS:0001		DS:0009	
DS:0002		DS:00DA	C51589022010920
DS:0003		DS:000B	
DS:0004		DS:000C	
DS:0005		DS:000D	
DS:0006		DS:00E	
DS:0007		DS:00F	

* Program Code:

```
model small  
addl num  
model small  
addl num macro n1,n2,ret1  
mov ax,n1  
add ax,n2  
mov result,ax  
Endm  
  
data  
num1 dw 1234h  
num2 dw 4321h  
res dw ?  
  
code  
mov ax,0  
mov ds,ax  
add num1,num2,ax  
ends  
end
```

* Practical Related Questions:

- 1) Write advantages and disadvantages of macro.
→ The advantage of macro is that it reduces the time taken for control transfer or in the case of function. The disadvantage of it is here the entire code is substituted so the programs length if a macro is called several times.
- 2) State the function of directive MACRO and ENDM.
→ MACRO - It is used to declare a Macro
ENDM - It is used to End Macro.

* Exercise

- 1) Write ALP using macros to perform following operations.
 $Z = (A+B) * (C+D)$

model small

```
Sum - num Macro n1,n2,res  
mov al,n1  
add al,n2  
mov result,al  
Endm
```

Date _____

delta

on 26 1

B-club 2

C ch 3

• 1 234

51 - id. n. 3

• 1183

→ slow?

Son du?

- Code sample for writing test cases

mvn clean, @dash, -DskipTests -DskipITs

more words, as well as some additional notes.

Sum - num a, b, n

Sum - num c, d, 32

May 21, 19

Melangez et mangez sans faim

mon 7. av

END

Mark obtained _____ Dated _____ Signature of _____
Teacher _____

Process Related	Product Related	Total
-----------------	-----------------	-------