**Ayush Goyal**

**190905522 CSE D 62**

**ES Lab 3 (Week 3) Programs on Arithmetic and Logical Instructions**

**Q1) Write an assembly language program to implement division by repetitive subtraction.**

**CODE:**

AREA RESET,DATA,READONLY

EXPORT \_\_Vectors

\_\_Vectors

DCD 0X10001000

DCD Reset\_Handler

ALIGN

AREA MYCODE,CODE,READONLY

ENTRY

EXPORT Reset\_Handler

Reset\_Handler

MOV R0,#10

MOV R1,#3

MOV R2,#0 ;Quo

MOV R3,#0 ;Rem

LDR R4, =QUO

LDR R5, =REM

UP CMP R0,R1

BCC Sto

SUBS R0,R1

ADD R2,#1;Quo

B UP

Sto MOV R3,R0

STR R2,[R4]

STR R3,[R5]

STOP B STOP

AREA MYDATA, DATA, READWRITE

QUO DCD 0

REM DCD 0

END

**OUTPUT:**

Graphical user interface, text, application, email

Description automatically generated

**Q2) Find the sum of ‘n’ natural numbers using MLA instruction.**

**CODE:**

AREA RESET,DATA,READONLY

EXPORT \_\_Vectors

\_\_Vectors

DCD 0X10001000

DCD Reset\_Handler

ALIGN

AREA MYCODE,CODE,READONLY

ENTRY

EXPORT Reset\_Handler

Reset\_Handler

LDR R0,=SRC

LDR R1,=DST

LDR R2,[R0]

MLA R3,R2,R2,R2

LSR R3,#1

STR R3,[R1]

STOP B STOP

SRC DCD 0x0000000A

AREA MYDATA, DATA, READWRITE

DST DCD 0

END

**OUTPUT:**

Graphical user interface, text, application, email

Description automatically generated

**Q3) Write an assembly language program to find GCD and LCM of two 8-bit numbers.**

**CODE:**

AREA RESET,DATA,READONLY

EXPORT \_\_Vectors

\_\_Vectors

DCD 0X10001000

DCD Reset\_Handler

ALIGN

AREA MYCODE,CODE,READONLY

ENTRY

EXPORT Reset\_Handler

Reset\_Handler

LDR R0, =SRC1

LDR R0, [R0]

LDR R1, =SRC2

LDR R1, [R1]

AGA CMP R0, #0

BEQ EXIT

UP CMP R1, R0

BHS CON

B SKIP

CON SUB R1, R1, R0

B UP

SKIP MOV R2, R0

MOV R0, R1

MOV R1, R2

B AGA

EXIT LDR R2, =GCD

STR R1, [R2]

LDR R0, =SRC1

LDR R0, [R0]

LDR R1, =SRC2

LDR R1, [R1]

MUL R0, R0, R1

LDR R1, =GCD

LDR R1, [R1]

LDR R2, =0

UP2 CMP R0, R1

BCC STO

SUB R0, R1

ADD R2, #1

B UP2

STO LDR R3, =LCM

STR R2, [R3]

STOP B STOP

SRC1 DCD 6

SRC2 DCD 8

AREA MYDATA, DATA, READWRITE

GCD DCD 0

LCM DCD 0

END

**OUTPUT:**

Graphical user interface, text, application

Description automatically generated

**Q4) Write an ARM assembly language program to convert 2-digit hexadecimal number into ascii format.**

**CODE:**

AREA RESET, CODE, READONLY

EXPORT \_\_Vectors

\_\_Vectors

DCD 0x10001000

DCD Reset\_Handler

AREA myCode, CODE, READONLY

ENTRY

EXPORT Reset\_Handler

Reset\_Handler

LDR R0, =DST

MOV R1, #0x12A

MOV R2, #0

MOV R3, #10

Up CMP R1, #0xA

BCC Sto

SUB R1, #0xA

ADD R2, #1

B Up

Sto ADD R1, #0x30

STRB R1, [R0], #1

MOV R1, R2

MOV R2, #0

CMP R1, #0xA

BCS Up

ADD R1,#0x30

STRB R1,[R0]

STOP B STOP

AREA MYDATA, DATA, READWRITE

DST DCD 0x0

END

**OUTPUT:**

Graphical user interface, text, application, email

Description automatically generated

**Q5) Write an ARM assembly language program to convert a 32-bit BCD number in the unpacked form into packed form.**

**CODE:**

AREA RESET, DATA, READONLY

EXPORT \_\_Vectors

\_\_Vectors

DCD 0X10001000

DCD Reset\_Handler

AREA MYCODE, CODE, READONLY

ENTRY

EXPORT Reset\_Handler

Reset\_Handler

LDR R0, =0x01020304

LDR R1, =0x0000000F

LDR R5,=DST

MOV R4, #4

UP AND R2, R0, R1

LSL R1, #4

LSR R0, #4

ORR R3, R2

SUBS R4, #1

BNE UP

STR R3,[R5]

STOP B STOP

AREA MYDATA, DATA, READWRITE

DST DCD 0

END

**OUTPUT:**

Graphical user interface, text, application, email

Description automatically generated

**THE END**