**Faculty of Computer and Information Systems**

**Islamic University of Madinah**

**Saudi Arabia**

**Computer Architecture and Assembly Language**

**CCS 2022**

**Section: 1539**

**Project**

***Matrix multiplication***

**Lecturer: Prof. Hatem El-Boghdadi**

**Lecturer: Bassam Al-Ahmdy**

**Submitted by**

**Mohammed Gamal 362049583**

**Ali Domlo 371003153**

**27 Rabi Al-Awwal, 1440**

**5 December, 2018**

**Table of contents:**

1. **Introduction**………………………………………….…………………………**1**
2. **Data Segment**…………………………………………………….……………….**2**
3. **Main Menu**………………………………………….….………………………....**3**
4. **Selecting number and dimensions of matrices**………...…………………...……**6**
5. **Entering data**……………..……………………….………………………….…**11**
6. **Performing the multiplication**…………………………...……………………..**14**
7. **Show the Result**……………………………………………....………………….**19**
8. **Exit**……………………………………………………………………………….**24**
9. **OutPut**…………………………………………………...…………………..…..**26**

**1. Introduction**

Matrix multiplication is one of the most important operations in many scientific calculations.

In this project we are interested in implementing the matrix multiplication operation in Assembly language. The operation is to be implemented for selected dimensions of multiplied matrices.

This project is an assembly language; because of that we are going to use the emulator (emu8086) for our application.

Data\_segment\_name segment

Enter\_matrix\_message dw 'Enter the number of matrices: $'

Enter\_row\_message dw 'Enter the number of row: $'

Enter\_column\_message dw 'Enter the number of column: $'

EnterData dw 'Matrix Number $'

EnterElements dw 'Enter elements (Row by Row) of Matrix Number : $'

Enter\_1 dw 'To Select number of matrices and dimensions press 1$'

Enter\_2 dw 'To Enter Data press 2$'

Enter\_3 dw 'To Perform multiplication and show results press 3$'

Enter\_4 dw 'To Exit press 4$'

ReturnMessage dw 'Press space bar to return to main menu$'

PrintMessage dw 'The Result Matrix:$'

Error\_Message dw '###Please Enter a number from ONE to FOUR only### $'

Exit\_Message dw '==)Exit the MATRIX MULTIPLICATION program $'

Error\_Enter dw 'Error: Number of row should equal the number of cloumns of previous matrix $'

Welcome\_Message dw '((\*\*\*+-\*-+\*\*\*)) Welcome to the MATRIX MULTIPLICATION program ((\*\*\*+-\*-+\*\*\*)) $'

NumOfMatrix dw ?

TempMat dw ?

TempRow db ?

TempCol db ?

ResultRow db ?

ResultColumn db ?

Result dw 200 dup(?)

TempResult dw 200 dup(?)

TempResultRow db ?

TempResultColumn db ?

NumOfRow db 256 dup(?) ;Define an Array that contains number of rows for each matrix

NumOfColumn db 256 dup(?) ;Define an Array that contains number of columns for each matrix

Data dw 2000 dup(?)

OffestOfMatrices dw 256 dup(?)

TempOffset dw ?

Sum dw ?

ZeroDigit dw ?

CheckNumOfRow db 0

Data\_segment\_name ends

Stack\_segment\_name segment

dw 16 dup()

Stack\_segment\_name ends

Code\_segment\_name segment

Main\_prog proc far

Assume SS:Stack\_segment\_name, CS:Code\_segment\_name, DS:Data\_segment\_name

Mov AX,Data\_segment\_name

Mov DS,AX

lea dx, Welcome\_Message ;((\*\*\*+-\*-+\*\*\*)) Welcome to the MATRIX MULTIPLICATION program ((\*\*\*+-\*-+\*\*\*)) "

Mov ah,9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

Start:

Mov DX,DS

Mov ES,DX

lea dx, Enter\_1 ;To Select number of matrices and dimensions press 1 "

Mov ah,9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

lea dx, Enter\_2 ;Print"To Enter Data press 2 "

Mov ah,9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

lea dx, Enter\_3 ; Print"To Perform multiplication and show results press 3"

Mov ah,9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

lea dx, Enter\_4 ;Print"To Exit press 4 "

Mov ah,9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

Mov ah,1 ;Take the PRESS from keyboard

int 21h

Mov ah,0

Sub al,30h

Mov bl,al

Mov ax,0b800h ;Clear Screen

Mov es,ax

Mov di,0

Mov cx,80\*25

Mov ax,0720h

Rep stosw

Mov dl, 0 ;Column

Mov dh, 0 ;Row

Mov bh, 0 ;Display page

Mov ah, 02h ;SetCursorPosition

int 10h

CMP bl,1

Je PRESS\_1

CMP bl,2

Je PRESS\_2

CMP bl,3

Je PRESS\_3

CMP bl,4

Je PRESS\_4

lea dx, Error\_Message ;Print "###Please Enter a number from ONE to FOUR only### "

Mov ah,9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

Jmp Start ;Return to the beginning of the code

PRESS\_1:

;Enter number and dimensions of matrices\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

Mov dl, 0dh ;Move to starting of line

Mov ah,2

Int 21h

lea dx, Enter\_matrix\_message ;Enter the number of the matrix: "

Mov ah,9

Int 21h

Mov ah,1 ;Take the number of the matrix from keyboard

int 21h

Mov ah,0

Sub al,30h

Mov NumOfMatrix,ax

Mov dl, 0ah ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0dh ;Move to starting of line

Int 21h

Mov TempMat,0 ;Set TempMat by 0

MatData:

Lea DX, EnterData ;Print "Matrix Number "

Mov AH,9

Int 21h

Mov DX,TempMat ;Print number of matrix

Add Dl,31h

Mov AH,2h

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

;ENTER NUMBER OF ROW

EnterRow:

lea dx, Enter\_row\_message ;Enter the number of row: "

Mov ah,9

Int 21h

Mov ah,1 ;Take the number of the row from keyboard

int 21h

Sub al,30h ;Find the value of row number

CMP TempMat,0 ;Check for first matrix

JNE Check

JMP StoreRow

Check:

CMP al,CheckNumOfRow ;check if number ofrow equal the number of cloumns of previous matrix

JNE DimensionError

JMP StoreRow

DimensionError:

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

lea DX,Error\_Enter ;Print "Error: Number of row should equal the number of cloumns of previous matrix"

Mov Ah, 9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

JMP EnterRow

StoreRow:

lea bx,NumOfRow

Add bx,tempmat

Mov [bx],al

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

;ENTER NUMBER OF COLUMN

lea dx, Enter\_column\_message ;Enter the number of column: "

Mov ah,9

Int 21h

Mov ah,1 ;Take the number of the column from keyboard

int 21h

Sub al,30h

lea bx,NumOfColumn

Add bx,tempmat

Mov [bx],al

Mov CheckNumOfRow,al

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

Inc TempMat

Mov AX,TempMat

Cmp NumOfMatrix,AX

JA MatData

Lea DX,ReturnMessage ;Print "Press any key to return to main menu"

Mov ah, 9

Int 21h

Press\_bar1:

Mov ah,1 ;Stop the program until the user press any key

Int 21h

Cmp al,32 ;Check if the user press space bar

JNE Press\_bar1

Mov ax,0b800h ;Clear Screen

Mov es,ax

Mov di,0

Mov cx,80\*25

Mov ax,0720h

Rep stosw

Mov dl, 0 ;Column

Mov dh, 0 ;Row

Mov bh, 0 ;Display page

Mov ah, 02h ;SetCursorPosition

int 10h

Jmp Start

PRESS\_2: ;Entering the data\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

Mov DX,DS

Mov ES,DX

Lea DI,Data

Lea CX,OffestOfMatrices

Mov TempOffset,CX

Mov TempMat,0

MatElements:

Push DI ;Store the offset of matrix

Mov AX,DI

Mov DI,TempOffset

Add DI,TempMat

Add DI,TempMat

Stosw

Pop DI

Mov TempRow,0 ;Determine number of Rows

Lea BX, NumOfRow

Add BX, TempMat

Lea DX, EnterElements

Mov AH,9 ;Print "Enter elements of Matrix Number "

Int 21h

Mov DX,TempMat ;Print matrix number

Add Dl,31h

Mov AH,2h

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

Int 21h

Mov dl, 0ah ;Move to starting of line

Int 21h

Row:

Mov TempCol,0 ;Determine Column number

Lea DX, NumOfColumn

Add DX, TempMat

Mov BP,DX

Column:

Mov dl, 8 ;Print backsapce

Mov ah,2

Int 21h

Mov AH,1 ;Enter the element

int 21h

Mov AH,0

Sub AL,30h ;Find the value of the element

Stosw

Mov dl, 9 ;Print Space

Mov ah,2

Int 21h

Inc TempCol ;Increament Column number

Mov AL,DS:[BP]

CMP AL,TempCol

JA Column

Mov dl, 0dh ;Move to New line

Mov ah,2

int 21h

Mov dl, 0ah ;Move to starting of line

int 21h

Inc TempRow ;Increament Row number

Mov AL,[BX]

CMP AL,TempRow

JA Row

Inc TempMat

Mov AX,TempMat

Cmp NumOfMatrix,AX

JA MatElements

Lea DX,ReturnMessage ;Print "Press any key to return to main menu"

Mov ah, 9

Int 21h

Press\_bar2:

Mov ah,1 ;Stop the program until the user press any key

Int 21h

Cmp al,32 ;Check if the user press space bar

JNE Press\_bar2

Mov ax,0b800h ;Clear Screen

Mov es,ax

Mov di,0

Mov cx,80\*25

Mov ax,0720h

Rep stosw

Mov dl, 0 ;Column

Mov dh, 0 ;Row

Mov bh, 0 ;Display page

Mov ah, 02h ;SetCursorPosition

int 10h

Jmp Start

PRESS\_3:

;MULTIPLICATION\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

Mov DX,DS

Mov ES,DX

Lea SI,Data

Lea DI,TempResult

Mov TempMat,0

Mov TempRow,0 ;Determine Row number

Lea BX, NumOfRow

Add BX, TempMat

MoveRow:

Mov TempCol,0 ;Determine Column number

Lea DX, NumOfColumn

Add DX, TempMat

Mov BP,DX

MoveColumn:

Movsw

Inc TempCol ;Increament Column number

Mov AL,DS:[BP]

CMP AL,TempCol

JA MoveColumn

Inc TempRow ;Increament Row number

Mov AL,[BX]

CMP AL,TempRow

JA MoveRow

Lea BX,NumOfRow ;Set number of Rows in Result matrix

Mov AL,[BX]

Mov ResultRow, AL

Multiply:

Lea BX, NumOfColumn ;Set number of Columns in Result matrix

Add BX, TempMat

Mov AL, [BX]

Mov ResultColumn, AL

Inc TempMat

Mov TempResultRow,0

Lea BX, NumOfRow ;Determine number of Rows

Add BX, TempMat

Lea DX, NumOfColumn ;Determine number of Columns

Add DX, TempMat

Mov BP,DX

FirstMatRow:

Mov TempCol,0

SecondMatCol:

Mov TempResultColumn,0 ;Determine Column number

Mov TempRow,0

Mov Sum,0

ColByRow:

Mov AL,TempResultRow ;Find the multiplicand(TempResultRow \* ResultColumn + TempResultColumn)

Mov AH,0

Mov CL,2

Mul CL

Mov CL,ResultColumn

Mul CL

Add AL,TempResultColumn

Add AL,TempResultColumn

Lea DI, TempResult

Add DI,AX ;The offset of Multiplicand will be in DI

Mov AL,TempRow ;Find the multiplier(TempRow \* DS:[BP] + TempCol)

Mov AH,0

Mov CL,2

Mul CL

Mov CL,DS:[BP]

Mul CL

Add AL,Tempcol

Add AL,Tempcol

Push BX

Lea BX,OffestOfMatrices ;The offest of starting of matrix

Add BX,TempMat

Add BX,TempMat

Mov SI,[BX]

Add SI,AX ;The offset of Multiplier will be in SI

Pop BX

Mov AX,[DI]

Mov CX,[SI]

Mul CX

Add Sum,AX

Inc TempResultColumn

Inc TempRow

Mov DL,ResultColumn

CMP DL,TempResultColumn

JA ColByRow

Mov AL,TempResultRow

Mov AH,0 ;Store Sum at (TempResultRow \* DS:[BP] + TempCol)

Mov CL,2

Mul CL

Mov CL,DS:[BP]

Mul CL

Add AL,TempCol

Add AL,TempCol

Lea DI,Result

Add DI,AX

Mov AX, Sum

Stosw

Inc TempCol

Mov Al,DS:[BP]

CMP AL, TempCol

JA SecondMatCol

Inc TempResultRow

Mov Al,ResultRow

CMP AL, TempResultRow

JA FirstMatRow

Lea SI,Result ;Copy Result to TempResult

Lea DI,TempResult

Mov AL,ResultRow

Mov CL,DS:[BP]

Mul Cl

Mov CL,AL

Mov CH,0

Rep Movsw

Mov AX,TempMat

Inc AX

Cmp NumOfMatrix,AX

JA Multiply

;Show Result \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

Mov CL,DS:[BP]

Mov ResultColumn, CL ;Set number of columns in Result

Lea DX, PrintMessage

Mov AH, 9

Int 21h

Mov dl, 0dh ;Move to New line

Mov ah,2

int 21h

Mov dl, 0ah ;Move to starting of line

int 21h

Mov TempRow,0

PrintRow:

Mov TempCol,0

PrintColumn:

Mov ZeroDigit, 0

Mov AL,TempRow

Mov AH,0 ;Printed elemnet at (TempResultRow \* ResultColumn + TempCol)

Mov CL,2

Mul CL

Mov CL,ResultColumn

Mul CL

Add AL,TempCol

Add AL,TempCol

Lea DI,Result

Add DI,AX

Mov AX,[DI]

Mov DX,0

Mov CX,10000; Find The value of ten thousands of the result

div CX

Mov BX,AX

Push DX

CMP BX, 0 ;Check if the result of division equal zero

JNE DisplayTenThousands

Mov DL, ' ' ;Print space

Mov AH, 2

Int 21h

JMP Thousands

DisplayTenThousands:

Add BL,30h ;Find the ASCII code of ten thousands

Mov DX,BX ;Display ten thousands

Mov AH,2

int 21h

Inc ZeroDigit

Thousands:

Pop AX

Mov DX,0

Mov CX,1000; Find The value of thousands of the result

div CX

Mov BX,AX

Push DX

CMP ZeroDigit, 0

JNE DisplayThousands

CMP BX, 0 ;Check if the result of division equal zero

JNE DisplayThousands

Mov DL, ' ' ;Print space

Mov AH, 2

Int 21h

JMP Hundreds

DisplayThousands:

Add Bl,30h ;Find the ASCII code of thousands

Mov DX,BX ;Display thousands

Mov AH,2

int 21h

Inc ZeroDigit

Hundreds:

Pop AX

Mov DX,0

Mov CX,100; Find The value of hundreds of the result

div CX

Mov BX,AX

Push DX

CMP ZeroDigit, 0

JNE DisplayHundreds

CMP BX, 0 ;Check if the result of division equal zero

JNE DisplayHundreds

Mov DL, ' ' ;Print space

Mov AH, 2

Int 21h

JMP Tens

DisplayHundreds:

Add Bl,30h ;Find the ASCII code of hundreds

Mov DX,BX ;Display ten hundreds

Mov AH,2

int 21h

Inc ZeroDigit

Tens:

Pop AX

Mov DX,0

Mov CX,10; Find The value of tens of the result

div CX

Mov BX,AX

Push DX

CMP ZeroDigit, 0

JNE DisplayTens

CMP BX, 0 ;Check if the result of division equal zero

JNE DisplayTens

Mov DL, ' ' ;Print space

Mov AH, 2

Int 21h

JMP Ones

DisplayTens:

Add Bl,30h ;Find the ASCII code of tens

Mov DX,BX ;Display tens

Mov AH,2

int 21h

Ones:

Pop BX

Add Bl,30h ;Find the ASCII code of ones

Mov DX,BX ;Display ones

Mov AH,2

int 21h

Mov dl, 9 ;Print tab

Mov ah,2

Int 21h

Inc TempCol ;Increament Column number

Mov AL,ResultColumn

CMP AL,TempCol

JA PrintColumn

Mov dl, 0dh ;Move to New line

Mov ah,2

int 21h

Mov dl, 0ah ;Move to starting of line

int 21h

Inc TempRow ;Increament Row number

Mov AL,ResultRow

CMP AL,TempRow

JA PrintRow

Lea DX,ReturnMessage ;Print "Press any key to return to main menu"

Mov ah, 9

Int 21h

Press\_bar3:

Mov ah,1 ;Stop the program until the user press any key

Int 21h

Cmp al,32 ;Check if the user press space bar

JNE Press\_bar3

Mov ax,0b800h ;Clear Screen

Mov es,ax

Mov di,0

Mov cx,80\*25

Mov ax,0720h

Rep stosw

Mov dl, 0 ;Column

Mov dh, 0 ;Row

Mov bh, 0 ;Display page

Mov ah, 02h ;SetCursorPosition

int 10h

Jmp Start

PRESS\_4: ;Exit the program\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

lea dx, Exit\_Message ;==)Exit the program "

Mov ah,9

Int 21h

Mov dl, 1h ;Print character

Mov ah,2

Int 21h

Mov AX,4c00h

Int 21h

Main\_prog endp

Code\_segment\_name ends

end main\_prog

Mov AX,4c00h

int 21h

Main\_prog endp

Code\_segment\_name ends

end main\_prog

***OUTPUT:***











