**CSC 3101 - FALL 2016**

**Lab Assignment 10**

Please read the requirements carefully. Any missing point will result in a grade reduction. This assignment has to be submitted to your **blackboard in class** in this **Word file**. Hard copy is **not accepted**.

**Question:** **(100 pts.)**

Please initialize an array [5, 0, 12, 3, 8, 21, 1, 14, 2, 16], and do the following operations on this array. You need to use subroutine and branch to finish this question. There should be 3 major subroutines in the program, one for each requirement given below:

1. In the 1st subroutine, load the input from keyboard. Decide whether the input is 1 or 2, and ask to load another input if it’s neither 1 nor 2. Store the value only when it’s 1 or 2 and continue to part b); (20 pts.)
2. In the 2nd subroutine, decide whether the elements in the initialized array are odd or even, and compare the results with the value provided in part a). That is, if the stored input is 1 in part a), use the 3rd subroutine to print out **ONLY** the elements with **odd values** decided by the program (5, 3, 21, 1). If the stored input is 2 in part a), use the 3rd subroutine to print out **ONLY** the elements with **even values** decided by the program (0, 12, 8, 14, 2, 16); (60 pts.)
3. In the 3rd subroutine, print out the required value in part b) on separate lines. Notice that this 3rd subroutine is basically just used as a function to be called every time when it’s need. (20 pts.)

(Please attach your code and screenshots in this file for submission.)

**CODE BELOW**

\*-----------------------------------------------------------

\* Program : Lab 10

\* Written by : Caleb Latimer

\* Date : 11/16/2016

\* Description: Subroutine + jump

\*-----------------------------------------------------------

ORG $1000

START:

MOVE.B #1,D2 ; LOAD 1 IN D2

MOVE.B #2,D3 ; LOAD 2 IN D3

MOVE #0,D6 ; LOADS 0 INTO D6, WILL FUNCTION AS A COUNTER

FIRST

LEA PROMPT, A1 ;1st subroutine PRINT MESSAGE

MOVE.B #14,D0

TRAP #15

MOVE.B #4,D0 ;get the input

TRAP #15

CMP.B D1,D2 ;compare the input to #1

BEQ PRELOAD ;IF 1 MOVE TO NEXT

CMP.B D1,D3 ;compare the input to #2

BEQ PRELOAD ;IF 2 MOVE TO NEXT

LEA TAUNT, A1 ;OTHER WISE PRINT ERROR AND START OVER

MOVE.B #14,D0

TRAP #15

BRA FIRST

PRELOAD

LEA ARRAY, A2 ;2nd subroutine, access the array aND STORE IT IN A1

MOVE D1,D7 ; STORES INPUT FROM FIRST INTO D7 FOR LATER USE

CLR.L D2 ; CLEARS D2

CLR.L D3 ; CLEARS D3

SECOND

MOVE.B (A2)+, D1 ;get the value of elements to D1, AND INCREMENT THE ARRAY TO THE NEXT ELEMENT

CMP #1,D7 ; DETERMINE IF EARLIER INPUT IS 1 IF SO CHECK FOR ODD VALUE IN D1

BEQ CHECKODD

CMP #2,D7 ;DETERMINE IF EARLIER INPUT IS 2 IF SO CHECK FOR EVEN VALUE IN D2

BEQ CHECKEVEN

INCREMENT

ADD #1,D6 ; ADD 1 TO D6

CMP.B #10,D6 ; COMPARE D6 TO 9

BNE SECOND ; LOOP HAS NOT COMPLETED SO START AGAIN , LOOP WILL COMPLETE WHEN D6 = 9, MAX NUMBER OF ELEMENTS IN ARRAY IS 10 AND WE START AT 0.

BEQ END

CHECKODD

MOVE D1,D2

AND #1,D2

BNE LAST

BEQ INCREMENT

CHECKEVEN

MOVE D1,D3

AND #1,D3

BEQ LAST

BNE INCREMENT

LAST

MOVE.B #3,D0 ;3rd subroutine, display the value stroed in D1

TRAP #15

LEA NEWL, A1 ;add a new line after displaying

MOVE.B #14,D0

TRAP #15

BRA INCREMENT ;return to the line where the 3rd subroutine is called

END

\* Variables and Strings

CR EQU $0D

LF EQU $0A

ARRAY DC.B 5, 0, 12, 3, 8, 21, 1, 14, 2, 16 ; Initialize the array

SIZE1 DC.W 10

PROMPT DC.B 'Please input 1 OR 2: ',0

TAUNT DC.B 'You did not input 1 or 2 please input 1 or 2 : ',CR,LF,0

MSG2 DC.B 'The input is not 1',CR,LF,0

MSGO DC.B 'Printing elements in the array: ',CR,LF,0

MSGT DC.B 'Array printed!',CR,LF,0

NEWL DC.B CR,LF,0

END START ; last line of source





