

UNIVERSITY OF CALIFORNIA, RIVERSIDE
Department of Computer Science and Engineering
CS61 – Machine Organization and Assembly Language
Homework 3

12

Given August 27, Due August 30, 2001

1. Implement the following high-level CASE statement using LC-2 assembly code. Do not use self-modifying code as shown in class.

```
while (true){
    cout << "Enter a number? "
    cin >> R1;
    select (R1){
    case 0:
        cout << "This is case 0" << endl;
        break;
    case 1:
        cout << "This is case 1" << endl;
        break;
    case 2:
        cout << "This is case 2" << endl;
        break;
    }
}
```

Answer

loop	.orig x3000	
	lea r0, prompt	;prompt "Enter a number?"
	trap x22	
	trap x20	;get ascii number
	trap x21	;echo number
	ld r1, asciidi	;convert from ascii number to decimal
	add r1, r0, r1	
	add r1, r1, r1	;multiply by 2 because each case uses 2 bytes
	ld r0, lf	;new line
	trap x21	
	lea r7, case0	;get base address of cases
	add r7, r1, r7	;r7 contains address of correct case
	ret	;jump to case
case0	lea r0, c0str	
	jmp	display
case1	lea r0, c1str	
	jmp	display
case2	lea r0, c2str	
display	trap x22	
	ld r0, lf	
	trap x21	

```

        jmp loop

prompt  .stringz "Enter a number? "
lf      .fill    x000a                ;line feed
asciidi .fill    xffd0
c0str   .stringz "This is case 0"
c1str   .stringz "This is case 1"
c2str   .stringz "This is case 2"
        .end

```

2. In the Tic-Tac-Toe game, we need to check for three adjacent X's to determine if player X has won. Write a subroutine to perform this check. The subroutine returns with the "P" (positive) flag set if X has won, and reset otherwise. The 3x3 Tic-Tac-Toe board is stored in 9 consecutive memory locations as define by the statement and picture

Board .BLKW 9, x0000

location 0	location 1	location 2
location 3	location 4	location 5
location 6	location 7	location 8

The contents of these locations are defined as follows:

0 = empty

1 = X in location

-1 = O in location