

**TA: Jade Cheng**  
**ICS 312**  
**Quiz Solution #3**  
**September 10, 2009**

**Indirect Addressing, Conditional Jump, Loop instruction (Note: This quiz is not counted)**

**Question:** Write the assembly code to calculate the summation of elements in an array with length 50, assuming the array contents are already there.

```
Answer:
    public sum
    .model small
    .data
List  dw 50 dup (?)
    .code
prog  proc
    :
    :                               ; code to populate list with numbers
    :
    mov     cx, 50                   ; cx is the loop counter.
    mov     ax, 0                   ; store the summation in ax.
    lea     bx, list                ; let bx point to the array.
looparray:
    cmp     cx, 0                   ; check if the end of array is reached.
    je     loopend
    dec     cx                       ; decrement the loop counter.
    mov     dx, [bx]
    cmp     dx, 0                   ; check if current element is 0.
    je     loopend
    add     ax, [bx]                ; add the current element to ax.
    add     bx, 2                   ; let bx point to the next element.
    loop   looparray
loopend:
    :
    :
prog  endp
end
```

```
Or:
    public sum
    .model small
    .data
List  dw 50 dup (?)
    .code
prog  proc
    :
    :                               ; code to populate list with numbers
    :
    mov     cx, 50                   ; cx is the loop counter.
```

```
        mov     ax, 0           ; store the summation in ax.
        mov     bx, 0           ; let bx be the index of the array.
looparray:
        cmp     cx, 0           ; check if the end of array is reached.
        je     loopend
        dec     cx             ; decrement the loop counter.
        mov     dx, list[bx]
        cmp     dx, 0           ; check if current element is 0.
        je     loopend
        add     ax, list[bx]    ; add the current element to ax.
        add     bx, 2           ; increment the index bx.
        loop   looparray
loopend:
        :
        :
prog    endp
        end
```