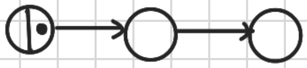
Comp. Arch. Unit 2

DATE: 25-03-2003

EXERCISE 1

linked ?

Write a function that, given a list on input, give the address of the last node as output.



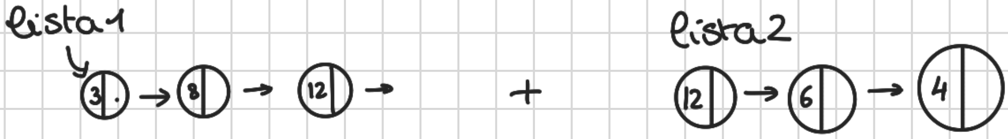
if zero, exit

```
coda:   lw  a1, 4(a0)           // a0 contains address of 1st value
        beq a1, zero, fine         with an offset of 4, you can
                                   find next address

ciclo:  mv  a0 , a1
        lw  a1 , 4(a1)          // a1 contains address of next
        bne a1, zero, ciclo        value, if you do +4, you get
        jalr  zero , ra, 0         next address
```

EXERCISE 2

Given two linked list, concatenate them.



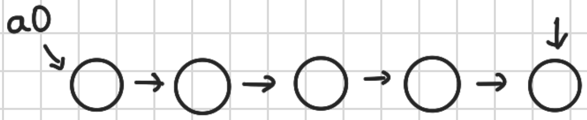
We can use the coda function.

.text

```
    lw a0, lista1       // lista1 contains address of 1st element
    jal coda
    lw t0, lista2       // t0 contains address of 1st el of 2nd list
    sw t0, 4(a0)        // saves t0 in the last node of 1st list
                           where there is 0.
```

EXERCISE 3

Given a list of integers, count how



n01 : . word 7, 0x10010010

n02 : . word 2 , 0

→ n03 : . word -1, n02

10010000 +4

10010004

n01 : . word 7, n02

n02 : . word -1, n03

n03 : . word 2, 0

→

+4

. word 7, 0x1001008

. word -1, 0x10010010

. word 2

7	0x10010000
0x10010010	0x10010004
→ 2	0x10010008
0	0x1001000c
-1	0x10010010 ←
0x10010008	0x10010014

7	0x10010000
0x10010008	0x10010004
-1	0x10010008
0x 10010010	0x1001000c
2	0x10010010
0x00	0x10010014