

# 1 luglio 2020 - parte 3: esercizio di programmazione

sino a 12 punti – è possibile consultare solamente il foglio consegnato con l'istruzione set MIPS - tempo: 60 minuti

Si scriva una procedura `media` in linguaggio Assembly MIPS32 che esegua una elaborazione degli elementi contenuti in un vettore di interi `veta` di dimensione nota `DIM` in cui sono memorizzati i valori numerici conseguenti al campionamento di un segnale analogico.

Il programma dovrà svolgere un'elaborazione sui dati contenuti in `veta` e scrivere il risultato in un vettore di interi `vetb` di uguale dimensione, effettuando le seguenti operazioni.

1) I primi due elementi in `vetb` rimangono invariati rispetto ai corrispondenti elementi in `veta`:

`vetb[0] = veta[0]`

`vetb[1] = veta[1]`

2) L'elemento in posizione  $i$ -esima di `vetb` (con  $1 < i < DIM-1$ ) è posto pari alla media aritmetica calcolata sui due valori precedenti di indice  $i-2$  e  $i-1$ , sul valore corrente di indice  $i$  e sul valore successivo di indice  $i+1$  in `veta`.

`vetb[i] = (veta[i-2] + veta[i-1] + veta[i] + vet[i+1]) / 4`

3) L'ultimo elemento di `vetb` rimane invariato rispetto al corrispondente elemento in `veta`:

`vetb[DIM-1] = veta[DIM-1]`

La procedura deve correttamente gestire i casi particolari in cui il vettore `veta` ha 1, 2 o 3 elementi.

Esempio:

posizione	veta	vetb
0	2	2
1	14	14
2	8	19
3	54	19
4	0	26
5	42	26
6	9	18
7	24	18
8	0	31
9	91	34
10	23	23

I parametri sono passati alla procedura attraverso i registri:

- `$a0` contiene l'indirizzo di `veta`
- `$a1` contiene l'indirizzo di `vetb`
- `$a2` contiene il numero di elementi di ciascun vettore, valore intero strettamente maggiore di 0.

Si lavori nell'ipotesi di non avere *overflow* durante i calcoli.

Di seguito un esempio di programma chiamante:

```
DIM = 11
.data
veta: .word 2, 14, 8, 54, 0, 42, 9, 24, 0, 91, 23
vetb: .space 4*DIM
.text
.globl main
.ent main
main: [...]
      la $a0, veta
      la $a1, vetb
      li $a2, DIM
      jal media
      [...]
      .end main
```

## Soluzione proposta

```
DIM = 11

.data

veta:      .word 2, 14, 8, 54, 0, 42, 9, 24, 0, 91, 23
vetb:      .space 4*DIM

.text
.globl main
.ent main
main:      subu $sp, $sp, 4
           sw $ra, ($sp)

           la $a0, veta
           la $a1, vetb
           li $a2, DIM

           jal media

           lw $ra, ($sp)
           addiu $sp, $sp, 4

           jr $ra
           .end main

.ent media
media:     lw $t1, ($a0)
           sw $t1, ($a1)
           beq $a2, 1, fine
           lw $t2, 4($a0)
           sw $t2, 4($a1)
           beq $a2, 2, fine
           addiu $a0, $a0, 8
           addiu $a1, $a1, 8

           lw $t3, ($a0)
           li $t0, 3
ciclo:     beq $t0, $a2, fineCiclo
           lw $t4, 4($a0)
           move $t5, $t4
           add $t5, $t5, $t1
           add $t5, $t5, $t2
           add $t5, $t5, $t3
           sra $t5, $t5, 2
           sw $t5, ($a1)
           addiu $a0, $a0, 4
           addiu $a1, $a1, 4
           move $t1, $t2
           move $t2, $t3
           move $t3, $t4
           addiu $t0, $t0, 1
           b ciclo

fineCiclo: sw $t3, ($a1)
fine:     jr $ra
           .end media
```