

Esercizi sulla programmazione in linguaggio assembly MIPS

Corso di Calcolatori Elettronici A.A. 2020-21

Esercizio 1

Scrivere il codice assembly che, dati due interi x e y , calcoli il valore di:

$$x - \min\{x, y\} + 1$$

scrivendo il risultato nel registro `$v0`. Assumere che x e y si trovino in memoria e che le etichette `X` e `Y` siano associate al loro indirizzo.

Esercizio 2

Scrivere il codice assembly che, dato un array di interi V e la sua dimensione, verifichi se V è ordinato in maniera decrescente.

Assumere che V e la sua dimensione N si trovino in memoria e che le etichette `ARR` e `N` siano associate, rispettivamente, all'indirizzo base di V e all'indirizzo di N .

Esercizio 3

Scrivere il codice assembly che, dato un array di interi V e la sua dimensione N , popoli V con i primi N elementi della successione di Fibonacci.

$$Fib(n) = \begin{cases} 1 & n \leq 2 \\ Fib(n-1) + Fib(n-2) & \text{altrimenti} \end{cases}$$

Esercizio 4

Scrivere il codice assembly che, dati due interi a e b , calcoli il loro massimo comun divisore $MCD(a, b)$ utilizzando l'algoritmo di Euclide:

Input: `a, b`

Se `b > a`:

 scambia `a` e `b`

Finche' `b != 0`:

`resto <- a mod b`

`a <- b`

`b <- resto`

Output: `a`