

Tutorato di Informatica - Soluzioni Foglio 2

28 Marzo 2025 - Foglio 2

Esercizio 1

Si consideri l'istruzione "j xx".

Domanda: Occupa più spazio in memoria la sua rappresentazione ASCII o l'istruzione macchina corrispondente dopo che è stata assemblata?

Esercizio 2

Interpretare il codice esadecimale 0x0232502A come istruzione MIPS, ed indicare:

- La sua rappresentazione binaria
- L'Op-code ed il formato (R-type, I-type o J-type)
- la rappresentazione simbolica dell'istruzione, completa del valore dei suoi parametri

Nota: usare l'appendice del libro a pagina A50

Esercizio 3

Indicare il formato dell'istruzione per effettuare la differenza tra due registri, ed esprimere esplicitamente l'operazione che effettua una sottrazione tra il valore nel registro \$3 e quello nel registro \$4, e deposita il risultato nel registro \$2:

- in forma simbolica
- in forma binaria
- in forma esadecimale

Esercizio 4

Stiamo eseguendo il seguente programma

Indirizzo	Istruzione (parziale)
0x00400000	add \$9, ...
0x00400004	sub \$14, ...
0x00400008	lw \$8, ...
0x0040000C	add \$7, ...

e ad un certo punto il program counter (PC) contiene il valore

0000 0000 0011 1111 1111 1111 1010 0100

inoltre sta per essere eseguita un'istruzione con op-code 000010. Come devono essere i bit rimanenti di questa istruzione, affinché dopo venga eseguita l'istruzione `lw $8, ...`?

Esercizio 5

Qual è l'indirizzo della più lontana cella di memoria in cui è possibile saltare (in avanti) con una istruzione `bne` se

PC=0010 0010 1100 0011 0110 0010 0100 0000 ?

Esercizio 6

Scrivere un programma che calcoli la somma del valore che è memorizzato nel registro \$17, e del valore memorizzato nella locazione di memoria che si trova 14 word più avanti dell'indirizzo specificato al registro \$16, e memorizzarsi il risultato 3 parole di memoria più avanti rispetto alla locazione attuale del secondo operando.

Nota: Vedi le istruzioni `lw` e `sw`, e ricorda che puoi usare la sintassi `n(x)` per indicare un indirizzo che abbia un offset di `n` byte rispetto all'indirizzo `x`

Esercizio 7

I valori relativi alle variabili della dimensione di una word `a`, `b`, `c`, `d` sono memorizzati di seguito in memoria a partire dall'indirizzo specificato dal contenuto del registro \$10. Scrivere la sequenza di istruzioni assembly che aggiunge la costante 10 alle variabili `a`, `b`, `c`, `d` e salva i nuovi valori delle variabili in memoria. N.B. Per la lettura dei valori della variabili successive alla prima, sarà possibile incrementare il valore dell'offset (di 4 byte per ciascuna word) oppure incrementare il contenuto del registro che contiene l'indirizzo base. La prima di queste opzioni NON permette di realizzare un ciclo mentre la seconda sì

Esercizio 8

In QtSpim, scrivere (manualmente) in memoria, nella sezione Data, la stringa seguente: "Hello world!". La stringa deve apparire leggibile nel giusto ordine nell'interfaccia del programma. NOTA: Usare la tabella ASCII (extended) per la codifica dei caratteri.

Esercizio 9

In QtSpim, inserire manualmente due valori nei registri \$t0 e \$t1. Poi scrivere un programma assembly che calcoli la somma dei due numeri presenti nei registri \$t0 e \$t1 e metta il risultato in \$t2.

Esercizio 10

Scaricare il programma `la_mia_prima_somma_Assembly.asm` da Moodle, caricarlo ed eseguirlo passo passo in QtSpim. Annotare, ad ogni passaggio (istruzione), il valore del PC e dei registri usati, ed inoltre monitorare cosa succede nelle sezioni `.text` e `.data`.

Esercizio 11

Completare il programma `ricerca_in_array.asm` che cerca il valore 5 all'interno di un array di 10 numeri interi. Farlo girare su QtSpim, passo passo e verificarne il funzionamento. Per verificare il funzionamento dell'intero programma, modificare il vettore perché contenga, o meno, il valore che si sta cercando.

Esercizio 12

Scrivere un programma in assembly che esegua le seguenti operazioni:

1. Leggere due numeri interi (num1 e num2) usando l'apposita syscall
2. Confrontare i due valori,
3. Stampare a schermo il più piccolo dei due.

dopodiché caricarlo su QtSpim e testarlo.

Consiglio: puoi partire dal file dell'esercizio precedente e prendere spunto; la tabella delle syscall ti può tornare utile.

Extra: stampa a schermo delle istruzioni per l'utente dove opportuno (consiglio: direttive `.data` e `.ascii`)