

Architettura dei Calcolatori a.a. 2012/13

Prova di Laboratorio - assembly MIPS: maxmin_array

9 Settembre 2013

Introduzione

Lo scopo di questa prova di laboratorio è lo sviluppo di un semplice programma nel linguaggio assembly del processore MIPS. Non è richiesta una particolare base di conoscenze algoritmiche, ma semplicemente un minimo di dimestichezza con la programmazione assembly.

Istruzioni

Cominciate facendo il login sulla macchina del laboratorio che vi è stata assegnata. Per il login occorre usare matricola e password dello *student portal*. Sul desktop troverete una cartella contenente i simulatori QtSpim e Mars. Lanciate ed utilizzate quello che preferite. Tutto il vostro codice (sia esso costituito da un singolo file, o da file multipli) andrà salvato nella cartella "mips" da creare sul drive H: .

Create un file `student-info.txt` con incluso il vostro nome e cognome e numero di matricola nella cartella "mips". Per maggior sicurezza, includete anche nome, cognome e matricola come commento, in testa ad ogni file sorgente. Alla fine della prova, i file saranno prelevati automaticamente dalla directory. Tutto quello che lascerete nella cartella mips sarà utilizzato per la valutazione. Salvare i vostri file altrove, o non indicare nome e cognome, porterà inevitabilmente all'annullamento della vostra prova. *Tutti i file all'esterno della cartella verranno cancellati automaticamente!!!*

Le specifiche

Dovete scrivere un programma assembly, il cui main allochi un array `arr` di 5 interi *sullo stack* (cioè, `arr` è una variabile locale del main, non una variabile globale). Una funzione `riempi_array` riceve il puntatore all'array come parametro e riempie ogni singola componente `arr[i]` con un valore ricevuto in input dall'utente. La funzione non ritorna alcun valore al main. Una seconda funzione `trova_max_min` riceve come parametro l'indirizzo dell'array, esegue un loop (*obbligatorio usare un loop!*) per individuare il max ed il min valore, e li ritorna in `$v0` e `$v1`. Il main stampa infine il max ed il min ricevuti da `trova_max_min`.

Per maggiori dettagli si veda l'output di esempio mostrato sotto. La chiamata delle due funzioni dovrà rispettare le convenzioni per il salvataggio dei registri. Utilizzate le system call SPIM/MARS per l'I/O e la terminazione del main. *È fondamentale utilizzare funzioni con scambio dei parametri, allocare l'array in area stack, e trovare il max ed il min mediante un loop.*

Suggerimenti

Potete leggere i numeri in input da tastiera utilizzando la system call 5. Per la stampa di stringhe e di interi potete usare le syscall 4 e 1, rispettivamente. Il seguente è un output di esempio:

```
Dammi il valore di arr[0]: 11
Dammi il valore di arr[1]: -227
Dammi il valore di arr[2]: 175858
Dammi il valore di arr[3]: 0
Dammi il valore di arr[4]: -5858959
```

```
Il massimo è 175858
```

```
Il minimo è -5858959
```

```
-- program is finished running --
Reset: reset completed.
```

Valutazione

Scrivere un programma funzionante, che faccia uso di due funzioni come richiesto nelle specifiche e che segua le convenzioni di salvataggio dei registri è strettamente necessario per essere ammessi a sostenere l'orale. In quella sede, si entrerà nel dettaglio della struttura del codice, dando una valutazione migliore a soluzioni "pulite" e ben commentate.