

Architettura dei Calcolatori, a.a. 2019/20

Prova di Laboratorio - assembly MIPS/RISC-V: cifre_pari

25 Gennaio 2021

Istruzioni

La prova si svolgerà completamente on-line. Dopo la connessione alla riunione Webex, vi verranno fornite in chat privata URL, login e password per la connessione ad un drive remoto in modalità WebDAV. Sul drive remoto troverete le versioni modificate di MARS (Mars4.5_as2.jar) e RARS (RARS1.4_as.jar). La prima è per i vecchi studenti (assembly MIPS), la seconda per gli studenti in corso (a.a. 2019/20 - assembly RISC-V). Lanciatela (o copiatela sul vostro desktop e lanciatela, è lo stesso). Create **subito** un file .asm col vostro cognome (es.: villano.asm) e dentro Nome, Cognome e Matricola e salvatelo **sul drive remoto** (non sul vostro PC!!!). A questo punto potete operare come al solito, lavorando sempre su quel file. Ogni tanto verrà eseguito un salvataggio automatico, ma non ve ne dovrete manco accorgere. Alla fine della prova, salvate il vostro file come di consueto.

La caduta del collegamento, l'utilizzo della versione "tradizionale" di MARS/RARS o comunque la mancanza dei file di auto-save sul drive remoto invalideranno la prova.

Le specifiche

Dovete scrivere un programma assembly, il cui `main` allochi *sullo stack* (cioè, come variabile locale del `main`) un intero. Il `main` legge in input l'intero e richiama una funzione che ricerca all'interno della rappresentazione decimale del numero (cioè del numero scritto normalmente, con cifre che vanno da 0 a 9) le cifre pari (0 è pari, per la cronaca) e le stampa.

Per maggiori dettagli si veda l'output di esempio mostrato sotto. Utilizzare per la lettura del numero da tastiera la system call per la lettura di interi e **non leggere il numero come una stringa** (sarebbe troppo facile!). La chiamata della funzione dovrà rispettare le convenzioni per il salvataggio dei registri. Utilizzate le system call MARS/RARS per l'I/O e la terminazione del `main`. ***È fondamentale utilizzare funzioni con scambio dei parametri, allocare l'intero in area stack e non leggere il numero come una stringa. Non è importante l'ordine in cui vengono stampate le cifre pari (basta che vengano stampate tutte).***

Suggerimenti

Il seguente è un output di esempio:

```
Dammi un intero:
1203456786

Le cifre pari nel numero dato sono:
2, 0, 4, 6, 8, 6

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