

Sistemi di Calcolo (A.A. 2023-2024)

Corso di Laurea in Ingegneria Informatica e Automatica
Sapienza Università di Roma



Compito (09/07/2024) – Durata 1h 30'

Inserire nome, cognome e matricola nel file `studente.txt`.

ISTRUZIONI PER STUDENTI DSA: svolgere a scelta due parti su tre.

Parte 1 (programmazione IA32)

In questo esercizio si chiede di tradurre in assembly una funzione che copia un array dato in input in uno spazio preallocato, fornito anch'esso in input, applicando la funzione `compute` agli elementi da copiare solamente se richiesto. Nella directory `E1`, si traduca in assembly IA32 la seguente funzione C scrivendo un modulo `e1A.s`:

```
short* cond_compute(short* array, unsigned* cond,
                    int n, short* out)
{
    int i = 0;

    for (i = 0; i < n; i++) {
        if (!cond[i])
            out[i] = array[i];
        else
            out[i] = compute(array[i]);
    }

    return out;
}
```

L'unico criterio di valutazione è la correttezza. Generare un file eseguibile `e1A` con `gcc -m32 -g`. Per i test, compilare il programma insieme al programma di prova `e1A_main.c`.

Non modificare in alcun modo `e1A_main.c`. Prima di tradurre il programma in IA32 si suggerisce di scrivere nel file `e1A_eq.c` una versione C equivalente più vicina all'assembly.

Parte 2 (programmazione di sistema POSIX)

Si scriva una funzione per la ricerca di contatti in una rubrica telefonica. Nello specifico, si scriva nel file `E2/e2A.c` una funzione con il seguente prototipo:

```
int cerca(const char * filename, const char * prefix,
          struct contatto ** arrayMatches)
```

che, dato in ingresso il nome `filename` del file contenente i dati di una rubrica telefonica ed una stringa `prefix`, restituisce in `arrayMatches` tutti i contatti aventi come prefisso nel nome la stringa `prefix` e come valore di ritorno il numero di contatti trovati. `arrayMatches` deve essere un array di elementi di tipo `struct contatto` definita nel file `e2A.h`. Gli elementi di `arrayMatches` devono comparire nello stesso ordine in cui compaiono nel file (in posizione 0 va il primo contatto che compare nel file il cui nome ha come prefisso `prefix`).

I contatti memorizzati in `filename`, con un record per riga, seguono il formato:

Mario Rossi_____+393297854258_

Dove i primi 31 byte corrispondono al nome completo e i successivi 15 al numero di telefono. Si noti che non vi sono separatori tra i campi di un record. I bytes in eccesso sono costituiti da padding rappresentato con il carattere `'_'`. I campi `nomecompleto` e `numero` delle struct `contatto` NON devono contenere caratteri di padding `'_'`.

Si può assumere che il file `filename` contenga al più 32 record.

Per i test, compilare il programma insieme al programma di prova `e2A_main.c` fornito, che **non** deve essere modificato. **Non** modificare il file `contatti1.txt` contenente la rubrica.

Parte 3 (quiz)

Si risponda ai seguenti quiz, inserendo le risposte (A, B, C, D o E per ogni domanda) nel file `e3A.txt`. Una sola risposta è quella giusta. Rispondere E equivale a non rispondere (0 punti).

Domanda 1 (Analisi delle prestazioni del software)

Di quanto è necessario ridurre una porzione di un programma che richiede il 35% del tempo di esecuzione per ottenere uno speedup sul programma di $\sim 1.32x$?

A	10%	B	50%
C	30%	D	70%

Motivare la risposta nel file `M1.txt`. **Risposte non motivate saranno considerate nulle.**

Domanda 2 (cache)

Si consideri una cache associativa a 2 vie con 4 linee da 64 byte ciascuna e politica di rimpiazzo LRU, inizialmente vuota. I blocchi dispari sono mappati sulle prime due linee, ed i blocchi pari sulle seconde due. Potendo scegliere tra più linee vuote, si usa la linea con indice più basso. Si ha inoltre un processo che accede in sequenza ai seguenti indirizzi di memoria (senza interruzioni): 670, 1612, 5454, 479, 3200, 144, 588.

Alla fine della sequenza di accessi, quali sono gli indici dei blocchi contenuti nelle 4 linee di cache? Il trattino indica che la linea di cache rimane vuota.

A	7, 9, 50, -	B	7, 9, 2, 50
C	2, 50, 7, 9	D	-, -, 2, 50

Motivare la risposta nel file `M2.txt`. **Risposte non motivate saranno considerate nulle.**

Domanda 3 (Pipeline)

Si consideri la seguente sequenza di istruzioni che vengono eseguite da una CPU con pipeline a 5 stadi (FDEMW):

```
1: movl $1, %eax
2: movl %eax, %ecx
3: movl $2, %edx
4: subl $1, %eax
5: testl %ecx, %ecx
6: jz E
```

Quale tipo di hazard si verifica al momento dell'esecuzione dell'istruzione 5?

A	Hazard strutturale	B	Hazard sui dati
C	Hazard sul controllo	D	Non si verifica nessun hazard

Motivare la risposta nel file M3.txt. **Risposte non motivate saranno considerate nulle.**

Domanda 4 (paginazione)

Si consideri un sistema di calcolo con spazio logico dei processi a 34 bit. Quanti byte sono ospitati in una pagina se la dimensione della tabella delle pagine è 8 MB? Si assuma che le entry della tabella delle pagine siano grandi ciascuna 32 bit.

A	32 KB	B	16 KB
C	8 KB	D	4 KB

Motivare la risposta nel file M4.txt. **Risposte non motivate saranno considerate nulle.**