

## Sistemi di Calcolo (A.A. 2018-2019)

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



### Compito (24/06/2019) – Durata 1h 45'

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

---

#### Parte 1 (programmazione IA32)

Nella directory E1, si traduca in assembly IA32 la seguente funzione C scrivendo un modulo `e1A.s`. La funzione inizializza una matrice quadrata `m` di lato `n` con valori forniti dalla funzione `value`.

```
#include "e1A.h"
void init_matrix(short** m, unsigned n) {
    unsigned i, j;
    for (i=0; i<n; ++i)
        for (j=0; j<n; ++j)
            m[i][j] = value(i, j);
}
```

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` fornito.

**Nota:** **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 vuole scrivere nel file E2/es2A una funzione `run` con il seguente prototipo:

```
void run(int* v, int n, int (*f)(int i))
```

Dato un array `v` di dimensione `n` e un puntatore a funzione `f`, la funzione `run`:

1. crea `n` processi dove il processo `i`-esimo (con `i` in  $[0, n-1]$ ) invoca `f(i)` ed usa il valore restituito dalla funzione come codice di exit;
2. attende la terminazione dei processi creati;
3. scrive in `v` i codici di terminazione degli `n` processi, in qualunque ordine (ad esempio, `v[3]` potrebbe contenere il codice di terminazione del primo processo creato, ecc.).

Per i test, compilare il programma insieme al programma di prova `e2A_main.c` fornito, che **non** deve essere modificato.

---

#### 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 (cache)

Si consideri una cache associativa a 2 vie con 4 linee da 64 byte ciascuna e politica di rimpiazzo LRU, inizialmente vuota. Potendo scegliere fra 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): 62, 413, 6200, 42, 916, 400, 520.

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.

<b>A</b>	8, 6, 15, 23	<b>B</b>	8, 6, -, -
<b>C</b>	6, 14, 96, -	<b>D</b>	6, 96, -, -

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

---

#### Domanda 2 (paginazione)

Si consideri un sistema di calcolo con spazio logico dei processi a 24 bit. Quanto deve essere grande una pagina affinché la tabella delle pagine occupi 1 KB? Si assuma che le entry della tabella delle pagine siano grandi ciascuna 32 bit.

<b>A</b>	32 KB	<b>B</b>	16 KB
<b>C</b>	64 KB	<b>D</b>	128 KB

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

---

#### Domanda 3 (ottimizzazioni)

Supponiamo di essere in grado di ridurre di un fattore 1.5x il tempo di esecuzione di una porzione di un programma mediante delle ottimizzazioni. Che percentuale del tempo complessivo deve prendere quella porzione per ottenere uno speedup complessivo per il programma pari a 1.2x?

<b>A</b>	50%	<b>B</b>	33%
<b>C</b>	67%	<b>D</b>	86%

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

---

#### Domanda 4 (processi)

Un sistema operativo time-sharing è progettato per dare l'illusione all'utente che più di un processo progredisca nella computazione sullo stesso core allo stesso tempo. Si supponga che la gestione di un context switch richieda 50 usec (microsecondi). Qual è la minima durata di time slice affinché il tempo speso nei context switch non superi l'1% del tempo totale impiegato dal sistema per eseguire processi in time-sharing?

<b>A</b>	4.95 msec	<b>B</b>	10.5 msec
<b>C</b>	100 usec	<b>D</b>	50 usec

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