

## Sistemi di Calcolo (A.A. 2020-2021)

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



### Compito (01/06/2021) – Durata 1h 30'

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`.

```
#include "e1A.h"

int count_tokens(char* str, const char* sep) {
    int cnt = 0;
    char* token = strtok(str, sep);
    while (token != NULL) {
        cnt++;
        token = strtok(NULL, sep);
    }
    return cnt;
}
```

Si assuma che `NULL` valga 0. 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 `load` con il seguente prototipo:

```
char* load(const char* filename, unsigned* size)
```

Dato un nome di file `filename` e un puntatore `size`, la funzione `load` carica da disco il contenuto del file in un nuovo buffer allocato dinamicamente che viene restituito dalla funzione e scrive in `*size` il numero di byte letti.

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 completamente associativa 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): 6200, 413, 62, 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, 0, 14	<b>B</b>	8, 6, 0, -
<b>C</b>	96, 6, 0, 14	<b>D</b>	6, 96, -, -

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

---

#### Domanda 2 (memoria virtuale)

Una sola delle seguenti affermazioni sulla memoria virtuale è **falsa**. Quale?

<b>A</b>	Lo spazio logico di memoria di un processo è distinto dallo spazio fisico di memoria	<b>B</b>	Gli indirizzi logici vengono mappati su indirizzi fisici grazie alla presenza di una tabella chiamata tabella delle pagine
<b>C</b>	Nella paginazione con bit di validità ogni pagina ha un bit che vale 1 se la pagina risiede in un frame e 0 altrimenti	<b>D</b>	E' possibile gestire spazi logici di grandi dimensioni (es. $2^{64}$ ) con una tabella delle pagine di qualche MiB e pagine di qualche KiB.

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 2x 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 1.5x?

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

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

---

#### Domanda 4 (processi)

Una sola delle seguenti affermazioni sui processi è **falsa**. Quale?

<b>A</b>	Un processo passa dallo stato running allo stato ready mediante preemption	<b>B</b>	Un processo diventa zombie se termina prima che il genitore ne abbia recuperato lo stato di terminazione con wait
<b>C</b>	Un processo passa dallo stato ready allo stato waiting invocando una system call	<b>D</b>	Il codice che segue l'invocazione di una execvp viene eseguito solo se la chiamata fallisce

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