

Sistemi di Calcolo (A.A. 2025-2026)

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



Compito (08/06/2026) – 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)

Nel campo della Data Science, la somma dei quadrati è una funzione chiave, utile a svolgere analisi statistiche su una serie di dati per misurarne la variazione, la dispersione e molto altro. In questo esercizio, si chiede di tradurre in assembly una funzione che, dato un insieme di numeri interi `data` e la sua dimensione `n`, calcola la somma dei quadrati dei soli valori compresi nell'intervallo aperto definito dai valori di soglia `min` e `max` e la scrive all'indirizzo puntato da `sum`. Nella directory `E1`, si traduca in assembly IA32 la seguente funzione C scrivendo un modulo `e1A.s`:

```
void threshold_squares_sum(int* data, int n, int min,
                           int max, int* sum)
{
    int i;
    *sum = 0;

    for (i = 0; i < n; i++) {
        if (data[i] < max && data[i] > min) {
            *sum += square(data[i]);
        }
    }
}
```

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 che aiuti l'utente nella scelta della squadra che parteciperà alla prossima competizione nazionale di CyberChallenge.it. Nello specifico, si scriva nel file `E2/e2A.c` una funzione con il seguente prototipo:

```
void selezione(const char * input, const char * output, int
num_players)
```

che, dato il nome `input` del file contenente informazioni sui partecipanti genera un file il cui nome è in `output`. Il file `input` contiene una riga per ogni partecipante con il nickname e il punteggio ottenuto nelle 4 categorie delle CTF (ovvero "cat1", "cat2", "cat3", "cat4") nel formato:

```
nickname punti_cat1 punti_cat2 punti_cat3 punti_cat4
```

I punteggi sono interi non negativi. Il programma deve identificare per ogni categoria il nickname del top player, ovvero il giocatore con il punteggio massimo in quella categoria. Un giocatore può risultare migliore in più categorie.

Il file di output generato deve contenere una riga per ogni categoria nel formato:

categoria,nickname

dove le categorie devono apparire nell'ordine cat1, cat2, cat3, cat4. Le righe in output nel file non devono contenere spazi e devono tutte terminare con \n.

Per i test, compilare il programma insieme al programma di prova e2A_main.c fornito, che **non** deve essere modificato. Nota: non modificare i file test*.txt che riportano esempi di file contenenti la lista partecipanti.

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)

Cosa rappresenta lo speedup di un programma ottimizzato rispetto all'originale?

A	Il numero di cicli CPU risparmiati	B	Il rapporto tra il numero di istruzioni eseguite prima e dopo
C	Il rapporto tra tempo originale e tempo ottimizzato	D	Il tempo medio tra due accessi in memoria

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

Domanda 2 (layout di memoria)

Si consideri l'output del comando cat/proc/17339/maps sul programma mem:

08048000-08049000	r-xp	00000000	00:19	55	/home/biar/Documents/temp/mem
08049000-0804a000	r--p	00000000	00:19	55	/home/biar/Documents/temp/mem
0804a000-0804b000	rw-p	00001000	00:19	55	/home/biar/Documents/temp/mem
0804b000-0a670000	rw-p	00000000	00:00	0	
0b3c7000-0b3e8000	rw-p	00000000	00:00	0	[heap]
f754d000-f754e000	rw-p	00000000	00:00	0	
f754e000-f76f6000	r-xp	00000000	08:01	415168	/lib/i386-linux-gnu/libc-2.19.so
f76f6000-f76f7000	---p	001a8000	08:01	415168	/lib/i386-linux-gnu/libc-2.19.so
f76f7000-f76f9000	r--p	001a8000	08:01	415168	/lib/i386-linux-gnu/libc-2.19.so
f76f9000-f76fa000	rw-p	001aa000	08:01	415168	/lib/i386-linux-gnu/libc-2.19.so
f76fa000-f76fd000	rw-p	00000000	00:00	0	
f771c000-f771f000	rw-p	00000000	00:00	0	
f771f000-f7720000	r-xp	00000000	00:00	0	[vdso]
f7720000-f7740000	r-xp	00000000	08:01	415161	/lib/i386-linux-gnu/ld-2.19.so
f7740000-f7741000	r--p	0001f000	08:01	415161	/lib/i386-linux-gnu/ld-2.19.so
f7741000-f7742000	rw-p	00020000	08:01	415161	/lib/i386-linux-gnu/ld-2.19.so
ffe38000-ffe59000	rw-p	00000000	00:00	0	[stack]

Quali dei seguenti può essere un puntatore verso la funzione main() del programma?

A	0x080483a0	B	0xffe43200
----------	------------	----------	------------

C	0xf75ff104	D	0x0804abc0
----------	------------	----------	------------

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

Domanda 3 (Pipelining)

Si consideri la seguente sequenza di istruzioni che vengono eseguite da una CPU con pipeline a 5 stadi (Fetch, Decode, Execute, Memory, Write-Back)

```
1: movl %(esp), %ebx
2: addl $0x10, %eax
3: movl %(ebx), %ecx
4: addl $0x42, %edx
```

Quale tipo di hazard si verifica al momento dell'esecuzione dell'istruzione nella linea 3?

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 (Puntatori e Valgrind)

Dato il seguente codice C:

```
void foobar() {
    int *arr = malloc(2 * sizeof(int));
    arr[2] = 42;
    return;
}

int main() {
    for(int i = 0; i < 100; i++)
        foobar();
    return 0;
}
```

Quale dei seguenti errori non è rilevato da un'esecuzione di Valgrind sul programma corrispondente?

A	Invalid read	B	Invalid write
C	Memory leak	D	Tutti i precedenti vengono rilevati

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