

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

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



Compito (16/09/2019) – 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 verifica se la stringa `s` è un prefisso della stringa `t`, cioè `t` inizia per `s`.

```
int is_prefix(const char* s, const char* t) {  
    while (*s && *t && *s == *t) { s++; t++; }  
    return *s == 0;  
}
```

L'unico criterio di valutazione è la correttezza. Generare un file eseguibile `e1A` con `gcc -m32 -g`. Per i test, compilare il codice 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 scriva nel file `E2/e2A.c` una funzione con il seguente prototipo:

```
void multi_get_env(const char** names, char*** values, int num)
```

La funzione prende come argomenti un array di stringhe `names` di lunghezza `num` e alloca un array di stringhe `*values` della stessa lunghezza dove `(*values)[i]` contiene il valore della variabile di ambiente `names[i]`, se la variabile è definita, e `NULL` altrimenti.

Suggerimento: usare la funzione `getenv`, consultando la documentazione `man`.

Generare un file eseguibile `e2A` con `gcc -g`. Per i test, compilare il codice insieme al programma di prova `e2A_main.c` fornito. **Nota importante:** fare attenzione ad allocare correttamente spazio per le stringhe e verificare la correttezza del codice usando `valgrind`.

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 (memoria virtuale)

Si consideri un sistema di memoria virtuale con uno spazio di indirizzi a 16 bit, pagine da 4 KB, e la seguente tabella delle pagine: {0x9, 0x3, 0xA, 0x7, 0xD, 0x1, 0x6, 0xF, 0xB, 0x2, 0xC, 0x5, 0x9, 0xE, 0x4, 0x8}. A quali indirizzi fisici corrispondono i seguenti indirizzi logici: 0x04AD, 0xE2AA, 0x6FE3?

A	0x04AE, 0xE2AC, 0x6FE7	B	0x94AD, 0x42AA, 0x6FE3
C	0x9DAD, 0x4AAA, 0x68E7	D	0x2A7B, 0xA9BB, 0x7EB1

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 32 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): 230, 56, 243, 67, 361, 182, 294.

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	2, -, 5, 9	B	7, 5, 9, 1
C	2, 11, 5, 9	D	2, -, 9, 5

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

Domanda 3 (permessi)

Che permessi (in notazione ottale) dovrebbe avere un file per essere accessibile in lettura, scrittura ed esecuzione dall'utente proprietario, lettura ed esecuzione dal gruppo proprietario, e solo esecuzione per tutti gli altri utenti?

A	0642	B	0751
C	0742	D	0754

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

Domanda 4 (pipelining)

Si consideri la seguente sequenza di istruzioni:

```
movl %edx, %eax
addl $1, %ebx
subw %ax, %cx
movl %ebx, %esi
```

Quanti cicli di clock vengono richiesti da una semplice pipeline a 5 stadi (Fetch, Decode, Execute, Memory, Write-Back) per completare tutte le istruzioni assumendo che gli hazard vengano risolti con stalli?

A	8	B	9
C	10	D	12

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