

NON SPEGNERE IL PC A FINE ESAME

Corso di Sistemi Operativi e Reti

Corso di Sistemi Operativi

Prova scritta di GIUGNO 2017

ISTRUZIONI

1. **Rinomina** la cartella chiamata "CognomeNomeMatricola" che hai trovato sul Desktop e in cui hai trovato questa traccia, sostituendo "Cognome" "Nome" e "Matricola" con i tuoi dati personali;
2. **Carica** tutto il materiale didattico che vorrai usare sul Desktop; puoi farlo solo nei primi 5 minuti della prova;
3. **Svolgi** il compito; lascia tutto il sorgente che hai prodotto nella cartella di cui al punto 1;
4. Quando hai finito lascia la postazione facendo logout,

senza spegnere il PC.

SALVA SPESSO il tuo lavoro

PER GLI STUDENTI DI SISTEMI OPERATIVI: si può sostenere solo uno dei due esercizi se si è già superata in un appello precedente la corrispondente prova. Il tempo a disposizione in questo caso è di 2 ORE.

NON SPEGNERE IL PC A FINE ESAME

ESERCIZIO 1 (Programmazione multithread. Punti: 0-20)

Bisogna implementare la classe `MatriceColorata` (il cui scheletro di codice viene fornito) secondo le seguenti specifiche:

-Una `MatriceColorata` è composta da una griglia di $N \times N$ caratteri; la matrice è inizialmente riempita di spazi;

-il metodo `stampa()` stampa a video il valore corrente della matrice;

-il metodo

```
int addSerpente (int lung, char c, boolean verticale, long delay)
```

aggiunge sulla matrice una striscia di `lung` caratteri di valore `c`, detta "Serpente" per default posta in orizzontale a partire dalla colonna 0 di un rigo scelto casualmente. Se il boolean `verticale` vale `true`, allora bisogna effettuare le stesse operazioni ma ponendo il "serpente" a partire dal rigo 0 di una colonna scelta casualmente, anziché una riga. `c` non può rappresentare uno spazio (" "), poiché questo simbolo viene usato per rappresentare la casella vuota.

Il metodo restituisce l'indice della riga/colonna in cui è stato inserito il serpente, o `-1` se nella posizione sorteggiata dovesse trovarsi, al momento dell'invocazione di `addSerpente`, parte di un serpente precedentemente inserito (e cioè la riga/colonna scelta deve contenere almeno `lung` spazi vuoti a partire dalla posizione 0 per potere inserire il nuovo serpente, altrimenti l'inserimento deve fallire).

Una volta inserito, un `Serpente` si muove di una casella per volta, a intervalli regolari di `delay` millisecondi, dall'alto verso il basso (se è un "serpente colonna") o da sinistra verso destra (se è un "serpente riga"), rientrando dall'estremità opposta una volta raggiunto il bordo destro/inferiore. Un serpente può muoversi solo se, nel momento in cui si sposta, la nuova posizione da occupare è libera (e cioè contiene uno spazio), altrimenti l'operazione deve essere rimandata di altri `delay` millisecondi.

Tutte le operazioni di stampa e aggiornamento della matrice devono essere thread-safe.

NON SPEGNERE IL PC A FINE ESAME

Esempi (bordo della matrice rappresentato col simbolo “|”):

| * * * * | → | * * * * | (serpente che rientra sul lato sinistro)

| |
| ° |
| * * * * ° | (serpente che non può andare verso destra a causa di
| ° | un altro serpente verticale di lunghezza 3)

| |
| % % % | (serpente di lunghezza 3 in posizione iniziale)
| |

NON SPEGNERE IL PC A FINE ESAME

NON SPEGNERE IL PC A FINE ESAME

CI SONO DEI PUNTI AMBIGUI NELLA TRACCIA? **COMPLETA TU**

È parte integrante di questo esercizio completare le specifiche date nei punti non esplicitamente definiti, introducendo o estendendo tutte le strutture dati laddove si ritenga necessario, e risolvendo eventuali ambiguità.

POSSO CAMBIARE IL PROTOTIPO DEI METODI RICHIESTI? **NO**

Non è consentito modificare il prototipo dei metodi se questo è stato fornito. Potete aggiungere qualsivoglia campo e metodo di servizio, e qualsivoglia classe ausiliaria, ma NON variare l'interfaccia dei metodi pubblici già specificati.

CHE LINGUAGGIO DEVO USARE? **JAVA 7 O SUCCESSIVO**

Il linguaggio da utilizzare per l'implementazione è Java. È consentito usare qualsiasi funzione di libreria di Java 7 o successivi.

MA IL MAIN() LO DEVO SCRIVERE? E I THREAD DI PROVA? **SOLO PER FARE IL TUO DEBUG**

Non è esplicitamente richiesto di scrivere un `main()` o di implementare esplicitamente del codice di prova, anche se lo si suggerisce per testare il proprio codice prima della consegna.

NON SPEGNERE IL PC A FINE ESAME

ESERCIZIO 2 (Linguaggi di scripting. Punti 0-10)

Lo scopo dell'esercizio è creare uno script (`file_utils.pl`) contenente una serie di funzioni di utilità per la gestione dei file.

Il comando potrà essere invocato in questo modo:

SYNOPSIS

```
file_utils.pl OPTION FILE [FILE_2]
```

Come si nota il parametro `OPTION` e il parametro `FILE` sono obbligatori, mentre il parametro `FILE_2` è opzionale (è obbligatorio solo per alcuni comandi, si veda sotto).

Come si può intuire i parametri `FILE` e `FILE_2` rappresentano i path di file nel sistema, mentre il parametro `OPTION` specifica il tipo di task di utilità che bisogna realizzare.

I possibili valori di `OPTION` e i task che bisogna eseguire per ciascuno di essi sono:

- `-w`
Utilizzando il comando `wc` bisogna identificare il numero di parole (`WORDS`) in `FILE` e visualizzarlo su Standard Output secondo il seguente formato:
Numero di parole: `WORDS`
- `-d`
Utilizzando il comando `diff` bisogna trovare le differenze tra `FILE` e `FILE_2` e visualizzarle su Standard Output nello stesso formato di `diff`
- `-s`
Utilizzando il comando `sort` bisogna ordinare le linee di `FILE` e salvarle in `FILE_2` (in questo caso specifico non bisogna verificare che `FILE_2` sia un path ad un file, vedi **Nota 3**)

Nota: è fondamentale verificare la presenza di tutti i parametri necessari

Nota 2: si può specificare una sola `OPTION`

Nota 3: si deve verificare che il valore di `FILE`, ed eventualmente di `FILE_2`, rappresentino effettivamente dei path a dei file. Questa verifica si può effettuare nel modo che si preferisce, per esempio utilizzando il comando `file`. In caso questa verifica dia esito negativo bisogna visualizzare su Standard Output solamente: (indipendentemente dall'opzione specificata)

```
Parametro errato
```

Nota 4: se l'`OPTION` specificata non corrisponde ad una di quelle disponibili, bisogna visualizzare su Standard Output solamente:

```
Parametro errato
```