

# Prova Pratica 058 - online

La prova consiste di due esercizi, il primo è un esercizio di programmazione concorrente, il secondo è un esercizio bash.

Avrete a disposizione circa 50 minuti per risolvere i due esercizi e consegnarle la soluzione richiesta.

Potete usare il materiale contenuto in <http://www.cs.unibo.it/~ghini/didattica/TREE4OS2021.tgz> che dovete già avere scaricato prima di questo esame

Non potete navigare in internet cercando informazioni all'esterno.  
Ovviamente, non potete comunicare con nessuno, in nessun modo.

Salvate i files spesso, per premunirsi in caso di crash.

# Esercizio Esame Pratica Online 161 - mondocane(1/2)

Ci sono  $N$  cani in un parchetto, ciascuno con indice da 0 a  $N-1$ . I cani, per istinto, si annusano le chiappe a vicenda. Ogni cane di **indice**  $I$  compie ripetutamente questa sequenza di operazioni:

- estrae un numero  $K$  casuale tra 0 e  $N+1$ ;
- se il numero  $K$  estratto è uguale all'indice  $I$  del cane stesso allora il cane estrae subito un nuovo numero;
- se il numero estratto  $K$  è maggiore o uguale ad  $N$  allora il cane aspetta che nessun cane gli stia annusando il culo e poi va a fare una pisciatina impiegando 6 secondi.
- se il numero  $K$  estratto è diverso dal proprio indice  $I$  ed è minore di  $N$  allora il cane di indice  $I$  verifica se il cane di indice  $K$  sta facendo pipì:
  - se fa pipì allora il cane  $I$  fa un giro per 2 secondi e poi estrae nuovo numero casuale.
  - se non sta facendo pipì allora il cane di indice  $I$  annusa il culo del cane di indice  $K$  per 3 secondi, poi abbaia al cane annusato per dirgli di avere completato l'operazione e poi va ad annusare margherite per 2 secondi per recuperare l'olfatto compromesso.

Un cane può essere annusato sempre, tranne che mentre sta facendo pipì.

Più cani possono annusare contemporaneamente lo stesso cane.

Ci sono  $N=7$  cani. Modellare ed implementare il sistema descritto, utilizzando dei thread POSIX per ciascuna figura (**cane**) ed avvalendosi delle opportune strutture dati per la sincronizzazione.

Utilizzare come base dell'esercizio i seguenti files, aggiungendo il codice necessario:

**mondocane.c**    **DBGpthead.c**    **DBGpthead.c**    **printererror.h**

Scrivere il Makefile per compilare e linkare i sorgenti. Inserire il necessario controllo di errore. In caso di errore grave, terminare il programma producendo un avviso a video.

Non è obbligatorio utilizzare, nel codice che aggiungerete voi, le funzioni **DBG\*** contenute nei files **DBGpthead.\*** ma è fortemente consigliato.

**SEGUE IN PAGINA SUCCESSIVA**

# Esercizio Esame Pratica Online **161** - **mondocane(2/2)**

## CONTINUAZIONE

-----

Si usa un vettore di interi, **NumCaniDietro[NUMCANI]** con tanti elementi quanti sono i cani. La posizione di indice K indica quanti sono i cani che, in quel momento, stanno annusando il culo del cane K.

Un cane può andare a fare pipì solo quando il valore della propria posizione nel vettore vale zero. Un cane quando fa pipì mette il valore a -1. Finito di fare pipì mette il valore a 0.

Oltre al vettore sopra descritto, potete usare solo queste due seguenti variabili globali:

`pthread_mutex_t mutex;`                      `pthread_cond_t cond;`

Per implementare le attese di un tempo variabile, usare la funzione **attendi(int min, int max)**. La funzione `attendi(int min, int max)` implementata nel file **mondocane.c** genera un numero casuale X compreso tra min e max ed attende X secondi prima di restituire il controllo al chiamante.

Le variabili globali già inserite sono inizializzate nel main.

Non potete aggiungere altre variabili globali.

# Esercizio Esame Pratica - 162 - **simmetrico**

Realizzare uno script `simmetrico.sh` che prende come argomento una stringa di caratteri alfanumerici (senza caratteri speciale e senza caratteri di quoting della bash) e verifica che la stringa sia simmetrica. In tal caso scrive in output "vero" altrimenti scrive in output falso.

Ad esempio sono simmetriche le stringhe `aa aba abcba zzzzzz z1z1z1z`

Realizzare lo script seguendo un approccio ricorsivo partendo dallo script base allegato.

Per facilitare la costruzione dello script secondo l'approccio ricorsivo si ricorda che:

- la stringa vuota è simmetrica.
- la stringa composta da un solo carattere è simmetrica.
- una stringa con due o più caratteri è simmetrica se il primo e l'ultimo carattere sono uguali e se la parte centrale della stringa è simmetrica.