

<http://tinyurl.com/thread5dic>

Thread (sia in C che in Java) -- Tempo a disposizione: 60 minuti

N.B.: Consegnare tutti i file sorgente prodotti (C e Java).

Scrivere in C e in Java un programma con due thread *TP* e *TD* che condividono una variabile *m* intera, che va inizializzata a 0.

I thread eseguono un ciclo infinito, comportandosi rispettivamente come segue:

TP attende 300 ms e successivamente controlla il valore *M* di *m*:

- se *M* è pari:
 - scrive *M* sulla standard output
 - genera un numero dispari casuale compreso tra 0 e 9 e lo memorizza in *m*
 - *TP* sveglia *TD*
- se invece *M* è dispari, *TP* si mette in attesa.

TD attende 200 ms e successivamente controlla il valore *M* di *m*:

- se *M* è dispari:
 - scrive *M* sulla standard output,
 - genera un numero casuale compreso tra 0 e 9 e lo memorizza in *m*;
 - *TD* sveglia *TP*
- Se invece *M* è pari, *TD* si mette in attesa.

N.B.

- in C, la chiamata `usleep(ut)` attende per *ut* microsecondi; in Java il metodo `sleep(mt)` della classe `Thread` attende per *mt* millisecondi.
- In C, ciascun thread esegue una funzione diversa `fun_p()` per *TP* e `fun_d()` per *TD*;
in Java, ciascun thread è istanza di una classe diversa, `ThreadP` per *TP* e `ThreadD` per *TD*.

Archiviare i file sorgente in un file denominato `nome_cognome_matricola.zip` e caricarli entro 60 minuti all'indirizzo:

<https://tinyurl.com/y73szemv>

Al termine dei 60 minuti non sarà più possibile caricare ulteriori files.