

Si vuole implementare in Java il seguente sistema:

- Sono presenti tre thread *Generator* che generano messaggi per il thread *Processor* che li elabora.
- I messaggi hanno una priorità 0, 1 e 2 (bassa, media, alta)
- I tre thread *Generator* generano ognuno messaggi ad una specifica priorità.
- I messaggi sono inseriti in tre code distinte a seconda della priorità.
- Ogni *Generator* produce 1000 messaggi rappresentati dai numeri interi $p*1000+i$ con $i=0..999$ e $p =$ priorità del messaggio
- Il thread *Processor* preleva prima dalla coda a priorità più alta, se vuota da quella a priorità intermedia e se vuota anche questa da quella a priorità bassa.
- I thread *Generator* dopo avere depositato il messaggio nella coda attende per un tempo T
- Il thread *Processor* dopo aver prelevato il messaggio lo stampa ed attende per un tempo T e quindi preleva un nuovo messaggio.
- Quando tutti i messaggi sono stati generati e sono stati tutti consumati dal *Processor* il programma deve terminare.

Implementare il sistema descritto usando i metodi sincronizzati per la sincronizzazione dei thread.

Il sistema soffre del problema della starvation, fornire una seconda implementazione che usi una tecnica che mitighi questo problema.