

Nome e cognome: .....Data.....04-02-2015.....

Matricola:.....

## 1. Specifiche generali

Scrivere un programma multithread che simuli uno zoo in cui si trova un gruppo di leoni ed un solo guardiano. In particolare il programma dovrà simulare il comportamento dei leoni che entrano in una capanna per mangiare e di un guardiano che distribuisce le porzioni di cibo ai leoni.

Se un leone entra nella capanna e non trova almeno una porzione di cibo sveglia il guardiano ed attende che questo prepari le nuove porzioni. Se, nel frattempo, altri leoni dovessero entrare nella capanna in cerca di cibo si metteranno in coda per attendere il servizio del guardiano. Al contrario, finché ci sono porzioni disponibili, i leoni entrano nella capanna e mangiano una porzione a testa contemporaneamente. Il programma dovrà garantire l'assenza di *race condition e deadlock*.

Il programma dovrà essere costituito da un oggetto "capanna" che sarà un oggetto condiviso tra 20 thread di tipo *leone* ed un solo thread di tipo *guardiano*. Il programma terminerà quando ogni leone avrà mangiato 10 volte.

## 2. Politica di gestione dei leoni in attesa

La politica di gestione delle attese è una politica di tipo FIFO ossia i leoni che si mettono in coda per il cibo dovranno essere risvegliati in ordine di arrivo.

## 3. Specifiche dell'oggetto Capanna

- L'oggetto Capanna sarà un oggetto condiviso tra i thread che rappresentano i leoni ed il thread che rappresenta il guardiano.
- Inizialmente nella capanna non ci sarà nemmeno una porzione di cibo pronta.
- Il tempo necessario per mangiare sarà pari a 200ms.
- L'oggetto Capanna esporrà il metodo pubblico invocato dai leoni "*richiediDiMangiare()*" ed il metodo pubblico invocato dal guardiano "*distribuiscePorzioni()*". Il primo sarà necessario per richiedere di mangiare e per verificare che ci sia almeno una porzione disponibile e, in caso contrario svegliare il guardiano e mettersi in attesa. Il secondo sarà invocato dal guardiano per rilasciare nuove porzioni di cibo.

## 4. Specifiche dell'oggetto Leone:

- L'oggetto leone sarà costituito da un thread e dovrà emulare un leone che, arrivando alla capanna richiederà di mangiare utilizzando il metodo pubblico "*richiediDiMangiare()*" descritto in precedenza. Se il leone trova una porzione disponibile inizia subito a mangiare simulando il tempo necessario sospendendosi per 200ms.
- Se non ci sono porzioni disponibili il leone sveglia il guardiano e si sospende in attesa che questo prepari le porzioni di cibo.
- Ogni leone dovrà essere caratterizzato da un tempo di arrivo alla capanna definito casualmente nell'intervallo [1,300] ms all'atto della creazione dell'oggetto.
- Il leone terminerà la propria esecuzione quando riuscirà a mangiare per 10 volte.

## **5. Specifiche dell'oggetto Guardiano:**

- L'oggetto guardiano dovrà emulare un guardiano che lavora presso uno zoo e che prepara il cibo per i leoni. Il guardiano, quando non ci sono richieste riposa senza utilizzare risorse. Non appena un leone lo sveglia il guardiano si mette al lavoro e produce  $N$  porzioni di cibo con  $N$  estratto casualmente in un intervallo  $[2,6]$ . Appena preparato il cibo il guardiano sblocca i leoni in attesa di mangiare e si rimette a riposo. Il guardiano terminerà quando tutti i leoni avranno terminato la propria esecuzione.