

# Sistemi Operativi T

## Prova di laboratorio

### 29 giugno 2022

#### Esercizio di Programmazione Concorrente in Java [punti 10]

Si consideri un sito speleologico, costituito da una grande grotta aperta alle visite del pubblico.

La grotta può essere visitata da due tipi di utenti:

- **adulti**;
- **bambini**.

Ogni visitatore (adulto o bambino) si comporta come segue:

1. **Entra** nella grotta
2. Visita la grotta impiegando un tempo arbitrario
3. **Esce** dalla grotta

Il regolamento del sito speleologico prevede che sia necessaria la presenza di **guide speleologiche** all'interno della grotta per supervisionare le visite ed eventualmente intervenire in caso di problemi. In particolare, detto **NG** il numero di guide nella grotta e **NV** il numero di visitatori nella grotta, deve essere sempre rispettata la relazione:

$$NG \geq NV/5 \quad (*)$$

Ogni guida può entrare e uscire dalla grotta, purchè la relazione (\*) sia sempre rispettata.

Ogni guida avrà quindi un comportamento ripetitivo; ad ogni ciclo attraverserà le seguenti fasi:

1. **Entra** nella grotta;
2. Permane nella grotta a supervisionare i visitatori per un tempo arbitrario;
3. **Esce** dalla grotta;
4. Sta all'esterno della grotta per un tempo arbitrario.

La grotta ha una **capacità limitata** MAX che esprime il numero massimo di **visitatori** che contemporaneamente possono stare all'interno della grotta (le guide non vengono conteggiate).

Realizzare un'applicazione concorrente in Java basata sul monitor nella quale Visitatori (adulti e bambini) e Guide siano rappresentati da thread concorrenti.

La sincronizzazione tra i thread dovrà tenere conto di tutti i vincoli dati ed inoltre dei seguenti criteri di priorità:

#### nell'accesso alla grotta:

- i visitatori adulti abbiano la precedenza sui bambini.

#### nell'uscita dalla grotta:

- i visitatori abbiano la precedenza sulle guide;
- i visitatori bambini abbiano la precedenza sugli adulti.