DIMES

Esame di **Sistemi Operativi** del Corso di Laurea in Ingegneria Informatica
Prova scritta del 16 giugno 2021 – Durata: 2 ore

Cognome e Nome:	Matricola:

TRACCIA "B"

Prova 1 (fino a 4 punti)

Dati i seguenti buchi di memoria: 150KB, 550KB, 200KB, 500KB, 700KB e 400KB (nell'ordine), illustrare come le strategie di allocazione della memoria

- **first-fit**,
- **worst-fit**

dispongono 5 processi che nell'ordine occupano: 210KB, 370KB, 120KB, 530KB e 334KB di memoria.

Prova 2 (fino a 4 punti)

Illustrare, anche attraverso schemi, il funzionamento della MMU nella realizzazione del mapping di indirizzi.

Prova 3 (fino a 4 punti)

Si descriva *(i)* il funzionamento della seguente applicazione Java, *(ii)* l'output che può produrre, e *(iii)* se l'applicazione termina.

```
public class Prova3_20210616_B {
    public static int var1 = 0, var2 = 0;
    public static Semaphore mutex = new Semaphore(1);

    public static void main(String[] args) throws InterruptedException {
        MyThread[] threads = new MyThread[7];
        for (int i = 0; i < 7; i++) {
            threads[i] = new MyThread();
            threads[i].start();
        }
        for (int i = 0; i < 7; i++) {
            threads[i].join();
        }
        System.out.println(Thread.currentThread().getName()+" "+Thread.currentThread().getState());
        System.out.println(var1+" "+var2);
    }//fine main
    static class MyThread extends Thread {

        public void run() {
            try {
                mutex.acquire();
                var1=var1+1;
                mutex.release();
                var2=var2+1;
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }//fine run
    }//fine class MyThread
}//fine classe Prova3
```

DIMES

Esame di **Sistemi Operativi** del Corso di Laurea in Ingegneria Informatica
Prova scritta del 16 giugno 2021 – Durata: 2 ore

Cognome e Nome:	Matricola:

Prova 4 (fino a 18 punti)

Si consideri un'agenzia di viaggi che organizza tour guidati a piedi del centro storico di *Firenze*.

Un turista arriva in piazza e si accoda al gruppo dei turisti interessati a partecipare alla visita. L'unica guida inizia la visita se sono presenti almeno 20 turisti. Se sono presenti meno di 20 turisti, la guida attende fin quando non si forma un gruppo di 20 turisti. Una volta formatosi il gruppo, attende 15 minuti (tempo dedicato a descrivere le attività della visita), e poi inizia il tour a piedi. I turisti che non sono riusciti ad entrare nel gruppo dei 20, attendono i turni successivi (risveglio FIFO).

La visita è strutturata in tre fasi: i) *prima-parte* che dura un tempo compreso tra 40 e 50 minuti, ii) *pausa*, che dura tra 10 e 20 minuti, e iii) *seconda-parte* che dura tra 40 e 50 minuti. I tempi esatti delle fasi vengono stabiliti dalla guida durante ogni tour. La guida riposa un'ora tra una visita e la successiva (lavora continuamente).

Si modelli il sistema descritto in Java, dove *Guida* e *Turista* sono dei *thread* che interagiscono tramite un oggetto *TourFirenze* che espone i seguenti metodi(*):

- **void attendiFormazioneGruppo()**: la *guida* attende la formazione del gruppo di turisti.
- **void visitaInizia():** la *guida* inizia il tour a piedi (avvisa i turisti dell'inizio del tour).
- **void visitaFine():** la *guida* finisce la visita della città.
- **void turistaInizia():** il *turista* attende l'inizio del tour. Si tratta di un metodo che blocca il turista fin quando non viene autorizzato dalla guida.
- **void turistaFine():** il *turista* finisce la visita. Il turista rimane bloccato fin quando la guida non finisce il tour.

Prova 4a (fino a 14 punti)

Si implementi la classe ***TourFirenze*** (astratta), ***Guida*** e ***Turista***, e una soluzione ***TourFirenzeLock*** che riproduca il funzionamento del problema sopra descritto utilizzando gli strumenti di mutua esclusione e sincronizzazione del package ***java.util.concurrent.locks.*** Si prega di scrivere le classi in quest'ordine: *TourFirenze(1), Guida(2), Turista(3)* e *TourFirenzeLock(4)*.

Prova 4b (fino a 4 punti)

Considerare la seguente variante. Il gruppo che inizia il tour può essere formato da più di 20 turisti (>=20). Durante la spiegazione del tour (15 minuti), tutti i turisti che si accodano in fila vengono autorizzati a partecipare al tour. Commentare, anche attraverso l'uso di frammenti di codice, come bisogna modificare le classi definite nella *Prova 4a*.

(*) Si possono aggiungere solo dei metodi di utilità (ad es. get e set)
N.B.: 1) Si chiede di commentare il codice e/o aggiungere delle stampe a video. Aggiungere un breve commento per ogni semaforo o condition.