

Note sullo svolgimento della prova

- Non è consentito l'uso di vecchi testi d'esame, libri, o appunti.
- Utilizzare NetBeans per editare, compilare ed eseguire i programmi.
- La cartella C:\prg contiene la documentazione delle Java API.
- Creare un file **Cognome_matricola.zip** contenente i soli file *.java* da consegnare.
- Al termine della prova, dopo il caricamento degli elaborati, il docente mostrerà una possibile soluzione. Dopo aver visto la soluzione, gli studenti avranno la possibilità di riprendere il proprio elaborato.

Esercizio

Un campeggio è dotato di n piazzole. Un cliente può prenotare con un'unica richiesta un numero di piazzole compreso tra 1 e n (estremi inclusi). Quando un cliente esegue una prenotazione con successo gli viene restituito un intero univoco che identifica la prenotazione stessa. Un cliente può disdire una prenotazione precedente, rendendo le piazzole di nuovo libere.

Realizzare una classe *Campeggio* dotata almeno dei seguenti metodi e costruttori:

- *Campeggio()*: crea un campeggio con 200 posti.
- *Campeggio(int n)*: crea un campeggio con n posti.
- *int prenota(int q, long t)*: prenota q piazzole; se il numero di piazzole disponibili è minore di q il cliente viene messo in attesa per un tempo non superiore a t millisecondi; in particolare, se prima di t millisecondi vengono liberati abbastanza posti per consentire la prenotazione del cliente in questione, allora il metodo esce dalla situazione di blocco il più presto possibile e completa la prenotazione; se invece non vengono liberate abbastanza piazzole allo scadere del tempo t il metodo lancia un'eccezione di tipo *PrenotazioneFallitaException* (da definire); il metodo restituisce l'identificatore univoco della prenotazione eseguita. Evitare attese attive.
- *void disdici(int p)*: disdice la prenotazione identificata da p (rendendo di nuovo libere le piazzole associate a tale prenotazione).

Realizzare inoltre una classe *Cliente* che

- prenota un certo numero di piazzole;
- nel 50% dei casi li disdice;
- termina la propria esecuzione.