

FONDAMENTI DI INFORMATICA (II e modulo B) (Prof. E. Lamma) – 15 giugno 2010
CORSO 2010 - PROVA PRATICA – LINGUAGGIO JAVA

NOTA PER LA CONSEGNA DEL COMPITO

Il compito deve essere stato **compilato ed eseguito**. E' importante che tutti i file e le classi si chiamino come richiesto. **Nel testo .java (come commento) va indicato il nome dello studente.**

Esercizio (da realizzare in un UNICO FILE .java)

Nella soluzione, prediligere il maggior riutilizzo di codice e la maggiore protezione possibile.

- Si realizzi una classe astratta **Tic**, che rappresenta un segnale trasmesso da un dispositivo collegato in rete. Tale classe ha come attributo di tipo intero **time** che rappresenta l'istante di tempo in millisecondi in cui viene ricevuto il segnale.

La classe è la seguente:

```
abstract class Tic {
    protected int time;
    abstract boolean check(int time); }
```

- Si realizzi poi un componente software **CheckedTic**, che deriva da **Tic**, che rappresenta la versione controllata del segnale di tic. Tale classe aggiunge l'attributo stringa **deviceName** e codifica un metodo costruttore a 2 argomenti (nome del dispositivo e l'istante di tempo). Implementa il metodo public **boolean check(int time)**. Tale metodo restituisce **true** nel caso in cui la differenza in valore assoluto tra il tempo in cui è arrivato il segnale corrente (attributo **time** dell'oggetto su cui è chiamato il metodo) e il tempo passato come parametro sia inferiore a 5000. Diversamente restituisce **false**. Usare il metodo **Math.abs()** per calcolare il valore assoluto.

Esempio:

```
int numero = -7;
int val_ass = Math.abs(numero); //val_ass contiene 7
```

Si realizzi poi un metodo **main** in una classe **Prova** che:

1. Legga, riga per riga, il file di testo **logs.txt** che è così strutturato: ogni riga contiene il nome del dispositivo e l'istante di tempo in cui è arrivato il segnale separate da “;”.

Esempio del contenuto:

```
A101;3278
B520;12345
A101;4277
```

2. Crei, per ogni riga del file, un oggetto **CheckedTic** usando il metodo **split** per ottenere i due valori (nome ed istante di tempo) da passare al costruttore. Il metodo **split** suddivide la stringa su cui è invocato in tante stringhe separate in base ad un carattere specificato e le restituisce sotto forma di vettore di oggetti di tipo **String**.

Usare il seguente codice per ricavare gli attributi dei nuovi oggetti da istanziare:

```
//line è la stringa letta dal file
String[] splitted = line.split(";");
String device = splitted[0]; //nome del dispositivo
int time = Integer.parseInt(splitted[1]); //istante di tempo
```

3. Verifichi tramite il metodo **check** invocato sugli oggetti creati che i tic dei dispositivi siano corretti, cioè che tra un istante e l'altro passino meno di 5000 msec. A tale scopo è necessario mantenere il valore del tic (time) precedente e poi passarlo come parametro al metodo **check**. Se **check** restituisce true, inserire l'oggetto in una lista (scegliere la migliore implementazione dalla JCF) denominata **logDevices**. Diversamente si stampi a video un messaggio di errore.
4. Una volta terminati la lettura del file e il riempimento della lista si stampi a video la dimensione e il contenuto della lista.

```
import java.io.BufferedReader;
```

```

import java.io.FileReader;
import java.util.ArrayList;
import java.util.List;

abstract class Tic {
    protected int time;
    abstract boolean check(int time);
}

class CheckedTic extends Tic {
    private String deviceName;
    public CheckedTic(String name, int time) {
        this.time = time;
        this.deviceName = name;
    }
    public boolean check(int time) {
        int diff = this.time - time;
        if (Math.abs(diff)<=5000) return true;
        return false;
    }
}

public class Prova {

    public static void main(String[] args) {
        try {
            List logDevices = new ArrayList();
            BufferedReader input = new BufferedReader(new FileReader("logs.txt"));
            String line = input.readLine();
            int prevTime = 0;
            while(line!=null) {
                String[] splitted = line.split(";");
                String device = splitted[0];
                int time = Integer.parseInt(splitted[1]);
                CheckedTic tic = new CheckedTic(device, time);

                if (tic.check(prevTime)) {
                    System.out.println("Alive time: " + time);
                    logDevices.add(tic);
                } else {
                    System.out.println("Error at time: " + time);
                }
                prevTime = time;

                line = input.readLine();
            }
            input.close();
            for (Object o: logDevices) {
                CheckedTic t = (CheckedTic)o; System.out.println(t);
            }
            System.out.println("Dimensione Lista: " + logDevices.size());

        } catch (Exception e) {
            System.out.println(e);
        }
    }
}

```