

**Prova Finale – Esempio di Compito**  
**Prova di Livello Intermedio**

---

COGNOME E NOME: \_\_\_\_\_ MATRICOLA: \_\_\_\_\_

**Tempo a disposizione: 4 ore**

**Esercizio n. 1: Taxi in C#**

E' necessario scrivere un programma in linguaggio **C#** che gestisca i dati relativi alle vetture di una compagnia di trasporti. La compagnia ha un nome ed un numero di telefono e possiede taxi e autobus; per ciascun taxi vuole rappresentare il numero di serie, il kilometraggio effettuato e il numero totale di minuti segnati dal tassametro; per ciascun autobus vuole rappresentare il numero di serie, il kilometraggio effettuato e la cilindrata. I dati relativi ai taxi sono tenuti in un file XML organizzato secondo un DTD opportuno. L'applicazione deve consentire di svolgere i seguenti caso d'uso:

**“Utente carica dati”**

- l'utente fornisce il nome di un file XML sul disco
  - l'applicazione carica i dati dal file XML
- Scenario alternativo: il nome del file è scorretto o ci sono problemi sul disco
- in questo caso l'applicazione segnala all'utente l'errore

**“Utente cerca veicoli con alto kilometraggio”**

- l'utente fornisce un valore del kilometraggio
- l'applicazione cerca e restituisce la lista dei veicoli (taxi o autobus) che hanno un kilometraggio superiore a quello specificato

**“Utente calcola minutaggio medio dei taxi”**

- l'applicazione calcola il numero medio di minuti contati dai tassametri dei taxi della compagnia
- Scenario alternativo: non sono stati forniti dati relativi ai taxi (numero di taxi = 0)
- in questo caso l'applicazione segnala che il calcolo non è possibile

Scrivere l'applicazione Java che effettua le operazioni elencate sopra secondo le seguenti specifiche:

- Sviluppare lo strato del modello e della persistenza. NON è necessario sviluppare interfaccia e controllo
- Sviluppare test di unità per i componenti di modello e persistenza
- Utilizzare un file di build di **NAnt** per le operazioni di costruzione

**Esercizio n. 2: Numero di giorni del mese**

E' necessario scrivere un'applicazione **Java Swing** per sapere quanti giorni contiene un mese dell'anno (28, 29, 30 o 31). L'applicazione deve consentire di svolgere i seguenti caso d'uso:

**“Utente calcola giorni del mese”**

- l'utente fornisce il nome del mese (es: "marzo")
  - l'applicazione calcola e visualizza i dati del mese di marzo
- Scenario alternativo: il nome del mese è scorretto
- in questo caso l'applicazione segnala all'utente l'errore

NOTA: leggere con attenzione le istruzioni relative ai nomi da attribuire alle cartelle alla pagina successiva

## Istruzioni per lo svolgimento della prova

A ciascuno studente sarà assegnato un computer. Per la durata della prova i computer saranno disconnessi dalla rete. L'accesso (login) alle macchine verrà effettuato preliminarmente dai docenti con un nome utente ed una password predisposta allo scopo, e non con il proprio nome utente usuale.

Durante le prove non sarà possibile uscire dall'aula, né comunicare oppure utilizzare appunti o libri o dischetti o qualsiasi altro dispositivo di memorizzazione di massa. Sarà invece disponibile la documentazione delle due piattaforme di riferimento (J2SE e .NET), oltre che gli strumenti di sviluppo utilizzati per le esercitazioni di laboratorio (compilatori, macchina virtuale, TextPad).

La documentazione del linguaggio Java è disponibile nella cartella **c:\j2sdk1.4.2\docs**. Per consultarla aprire il file **index.html** con un qualsiasi browser Web. La documentazione del framework .NET è disponibile a partire dal file **startHere.htm** nella cartella **C:\Programmi\Microsoft.NET\SDK\v1.1\**; aprire il file utilizzando Internet Explorer.

La documentazione del linguaggio Java è disponibile nella cartella **c:\j2sdk1.4.2\docs**. Per consultarla aprire il file **index.html** con un qualsiasi browser Web. La documentazione del framework .NET è disponibile a partire dal file **startHere.htm** nella cartella **C:\Programmi\Microsoft.NET\SDK\v1.1\**; aprire il file utilizzando Internet Explorer.

**JUnit** è già installato sulle macchine ed aggiunto al CLASSPATH; per eseguire il testRunner grafico è possibile utilizzare il comando **java junit.swingui.TestRunner <nomeDellaClasseDiTest>**

**Nunit** è disponibile nella cartella **c:\Programmi\NUnit V2.1**; per eseguire il testRunner grafico è sufficiente lanciare l'applicazione **Nunit-GUI**; per compilare le classi di test è necessario fare riferimento all'assembly del framework con la seguente sintassi **csc /r:"c:\Programmi\NUnit V2.1\bin\nunit.framework.dll" <classi>.cs**

Tutto il materiale disponibile durante la prova (classi, file di build, jar) è contenuto nella cartella **c:\lib\progOggettiII**

Il tutorial di Swing è disponibile in formato .zip nella cartella **c:\lib\progOggettiII**. Se è necessario, decomprimerlo all'interno della stessa cartella.

La soluzione dei quesiti proposti consisterà in una serie di file. **I file dovranno essere salvati esclusivamente in una cartella assegnata del disco C**. In particolare:

- gli studenti del **I turno** di ogni giornata dovranno salvare i propri file nella cartella **"c:\esami\POO\lappoggio1"**
- gli studenti del **II turno** di ogni giornata dovranno salvare i propri file nella cartella **"c:\esami\POO\lappoggio2"**

**Il codice Java ed il codice C# dovranno essere contenuti in cartelle separate. Nei nomi di entrambe le cartelle dovrà essere contenuto il cognome e l'iniziale del nome dello studente che li ha creati**

Esempio: lo studente Pasquale Rossi che partecipa alla prova nel **I turno**, creerà le seguenti cartelle:

- una cartella **c:\POO\lappoggio1\rossip-java** contenente tutti i package Java dell'esercizio n. 1 (ovvero: c:\POO\lappoggio1\rossip-java\it\unibas\xxx\Principale.java ecc.)
- una cartella **c:\POO\lappoggio1\rossip-csharp** contenente tutti i namespace C# dell'esercizio n. 2 (ovvero: c:\POO\lappoggio1\rossip-csharp\xxx\Principale.cs ecc.)

Al termine della prova lo studente dovrà accertarsi di aver salvato i file nella cartella assegnata e dovrà lasciare la sala senza spegnere la macchina. I docenti passeranno successivamente a recuperare i file dalle cartelle di ciascuna macchina.

**NOTA:** La correzione si baserà solo sui file prelevati dalla cartella **c:\POO\lappoggio1** per il I turno e **c:\POO\lappoggio2** per il secondo turno. Non saranno presi in considerazione altri file.