

1. Setup iniziale

Prima di iniziare, assicurati di avere:

- Java JDK (preferibilmente Eclipse Adoptium Temurin 17)
- Mercurial (versione 6.0 o superiore)
- Gradle (fornito nel repository, ma verificane il funzionamento)
- Un IDE (Visual Studio Code, Eclipse o IntelliJ IDEA)

2. Configurazione dell'ambiente

Se non l'hai già fatto, configura Mercurial:

```
hg config -e
```

Aggiungi o modifica queste righe nel file di configurazione:

```
[ui]
username = Nome Cognome <nome.cognome@studenti.unipd.it>

[extensions]
churn =
histedit =
mq =
rebase =
strip =
```

3. Clona il repository del corso (se non già fatto)

```
hg clone https://hg.sr.ht/~michelemauro/pdp2024-pub
cd pdp2024-pub
```

4. Verifica lo stato del repository

```
hg summary
```

Assicurati che il repository sia aggiornato:

```
hg pull
hg update default
```

5. Creazione del branch personale

Sostituisci "MATRICOLA" con il tuo numero di matricola:

```
hg branch MATRICOLAM
```

6. Crea il file delle versioni

```
.\gradlew -version > version.txt  
hg --version >> version.txt  
hg add version.txt  
hg commit -m "Versioni usate"
```

7. Aprire il progetto nell'IDE

Apri la cartella `pdp2024-pub` nell'IDE. È consigliabile aprire l'intero progetto dalla directory principale per accedere correttamente a tutti i file di progetto e test.

8. Implementazione della soluzione

Implementa le classi richieste per l'esercizio:

- `VideoSystem.java`
- `Printer.java`
- `Main.java`

Ricorda che i punti da modificare sono segnalati da commenti `TODO`.

9. Verifica il funzionamento

Esegui il programma per verificare che funzioni correttamente:

```
.\gradlew ex1
```

o in alternativa, esegui direttamente il `Main.java` dal tuo IDE.

Se l'output è simile a questo, la soluzione funziona correttamente:

```
_      13,50  
--      26,80  
...  
-----!!!!!---      193,13  
Esercizio completato. Punteggio finale: 212.63144
```

10. Formattazione del codice

Prima della consegna, applica la formattazione richiesta:

```
.\gradlew spotlessApply
```

11. Preparazione per la consegna

Assicurati che il tuo branch sia aggiornato con le ultime modifiche dal ramo default:

```
hg update MATRICOLAM
hg merge default
hg commit -m "Implementazione esercizio 1"
```

12. Crea il bundle con le tue modifiche

```
hg bundle -b MATRICOLAM MATRICOLAM.hg
```

13. Condividi il bundle con il professore

1. Carica il file `MATRICOLAM.hg` su Google Drive dell'account UniPD
2. Seleziona il file
3. Clicca su "Condividi"
4. Aggiungi `michele.mauro@unipd.it` come destinatario
5. Imposta le autorizzazioni su "Visualizzatore"
6. Conferma cliccando su "Fine"

Troubleshooting

Se il merge fallisce

```
hg commit -m "Modifiche prima del merge"
hg update MATRICOLAM
hg merge default
```

Se continua a fallire:

```
hg update --clean -r tip
```

Questo comando forza l'aggiornamento alla revisione più recente, ma perderai eventuali modifiche non salvate.

Se ci sono problemi di formattazione

Se `spotlessApply` trova errori di formattazione, li correggerà automaticamente. Dopo l'esecuzione, verifica nuovamente il funzionamento del codice.

Se la JVM non termina correttamente

Verifica che tutti i thread siano terminati correttamente. Nel `Main.java`, controlla:

1. Che la condizione del ciclo `do-while` sia corretta
2. Che tutti i componenti che non si fermano da soli (`score` e `printer`) vengano fermati esplicitamente

Se il comando `gradlew` non funziona

Assicurati di essere nella directory principale del progetto (`pdp2024-pub`) quando esegui i comandi `gradlew`.

Riepilogo dei passaggi chiave

1. Crea branch personale (`hg branch MATRICOLAM`)
2. Implementa le soluzioni richieste
3. Verifica il funzionamento (`.\gradlew ex1`)
4. Formatta il codice (`.\gradlew spotlessApply`)
5. Aggiorna e unisci con il ramo default (`hg merge default`)
6. Crea bundle (`hg bundle -b MATRICOLAM MATRICOLAM.hg`)
7. Condividi su Google Drive con `michele.mauro@unipd.it`

Ricorda che la consegna deve avvenire entro la data stabilita (23:59 del 22/4 secondo le slide).