



1 Normativa de l'activitat

1. El codi de cada mètode ha d'estar comentat descrivint en quines condicions és aplicable (precondició) i quin és el resultat final (postcondició).
2. Per definir les classes Jugador, Personatge i Equipament no podeu fer servir classes diferents a les que heu vist fins ara al laboratori o a teoria.
3. Disposareu de diversos jocs de prova públics. Per cada un d'ells hi haurà un arxiu d'entrada i un altre amb la sortida corresponent. Heu de respectar el format de les entrades i les sortides que apareixen en aquests arxius.
4. El que haureu de lliurar és un arxiu comprimit (.zip o .rar) amb tots els arxius *.hpp i *.cpp que siguin necessaris per compilar i muntar el vostre programa (NOMÉS aquests fitxers). Dins d'Atenea trobareu una Tasca on penjar el vostre arxiu. No s'acceptaran treballs enviats fora de termini ni enviats de cap altra manera.

Data límit: diumenge 18 d'octubre a les 23 h 59 min

5. Per valorar l'activitat es tindran en compte tant l'especificació com la implementació així com la superació de diverses execucions.
6. Per aquesta activitat, heu de formar equips de dues persones com a màxim.
7. En el cas de detecció de còpia entre dos o més equips, la normativa de l'EPSEVG determina que la qualificació tant de l'activitat com de l'assignatura serà de SUSPENS (0) per totes les persones integrants de tots els equips.

Les accions irregulars que poden conduir a una variació significativa de la qualificació d'un o més estudiants constitueixen una realització fraudulenta d'un acte d'avaluació. Aquesta acció comporta la qualificació descriptiva de suspens i numèrica de 0 de l'acte d'avaluació i de l'assignatura, sense perjudici del procés disciplinari que es pugui derivar com a conseqüència dels actes realitzats.



2 Disseny modular en C++:

Volem dissenyar un videojoc de rol on els jugadors poden crear-se personatges i cada personatge es pot equipar amb diferents elements. Per aquesta part del disseny del joc s'han definit el requeriments funcionals següents:

- Jugador: cal mantenir la informació de cada jugador com ara el nom, el login i el password.
- Personatge: cada jugador podrà tenir una sèrie de personatges (màxim 5) i de cadascun d'ells caldrà guardar el nom, l'ofici (Guerrer, Mag, Sacerdot, Paladí) i el gènere (masculí, femení, neutre).
- Equipament: cal mantenir tots els ítems que serviran per equipar als personatges. Cada ítem tindrà un nom, un tipus (Arma, Roba, Armadura, Talismà) i un pes.

Per representar aquesta informació i crear el sistema de definició de personatges farem servir 3 classes: Jugador, Personatge i Equipament. També caldrà desenvolupar un programa que provarà totes les funcionalitats previstes.

2.1 Primera tasca

La [primera tasca](#) d'aquesta activitat serà especificar les classes Jugador, Personatge i Equipament com es descriuen a continuació:

Un Jugador s'identifica amb un nom, té un login i el password. A efectes de simplificació, els passwords es guardaran sense encriptar a l'aplicació. Exemple de com es mostren les dades d'un jugador:

```
Nom: John  
Login: john71  
Password: J0hN71
```

Un Personatge s'identifica amb un nom, tindrà un ofici que només pot ser "Guerrer", "Mag", "Paladí" o "Sacerdot"¹, el gènere "Masculí", "Femení", "Neutre" i el pes màxim que pot portar en kilograms d'equipament (que no pot superar els 75Kg). Exemple de com es mostren les dades d'un personatge:

```
Nom: Diallos  
Tipus: Guerrer
```

¹ No farem servir accents per a evitar problemes de visualització de dades



Gènere: Neutre
Pes màxim: 50kg

Cada ítem d'Equipament tindrà un nom, que l'identifica, un tipus "Arma", "Roba", "Armadura", "Talisma") i un pes en kilograms (els ítems mai tindran un pes superior als 30Kg). Exemple de com es mostren les dades d'un ítem:

Nom: Destral
Tipus: Arma
Pes: 5Kg

Recordeu que cada jugador només pot crear 5 personatges i que el pes de tot l'equipament de cada personatge no pot superar el pes màxim que té especificat. Tampoc serà possible que un personatge es pugui equipar amb més de 6 ítems ni que hi hagi dos personatges amb el mateix nom, encara que siguin de jugadors diferents.

Un Joc s'identifica amb un nom i disposa d'un conjunt de jugadors i d'ítems d'equipament. Cada jugador té els seus personatges equipats amb totes les condicions que s'han especificat.

Per cada classe, haureu de definir: almenys un constructor, un destructor, modificadors dels atributs (quan sigui necessari), consultors per accedir als atributs i operacions de lectura i escriptura.

Recordeu que, en cridar un mètode, haureu de comprovar primer si se satisfà la precondition i, en cas contrari, haureu de decidir què ha de fer l'aplicació per tractar aquesta situació.

2.2 Segona tasca

La [segona tasca](#) serà implementar les classes seguint l'especificació que heu creat. Cal que testegeu cada classe individualment, per assegurar-vos que funcionen correctament. Per això caldrà crear abans petits programes principals que provin cada classe per separat i els corresponents jocs de proves unitaris.

És molt important que sigueu molt curosos i proveu extensament les classes abans de passar a la tercera tasca.

2.3 Tercera tasca

La [tercera tasca](#) serà implementar una aplicació simple que permeti gestionar les dades dels jugadors, els personatges i els ítems d'equipament que acabem de descriure. L'aplicació



oferirà la possibilitat de provar les operacions següents llegides per l'entrada estàndard:

1. Crear un jugador [opció **crearJugador**]

A l'hora de crear un jugador cal comprovar que no hi hagi cap altre jugador amb el mateix nom. Si se satisfà la condició, es crea el jugador i es mostra un missatge d'èxit, altrament el programa ha de mostrar un missatge d'error i el jugador no es crearà.

2. Crear un ítem d'equipament [opció **crearItem**]

A l'hora de crear un ítem d'equipament cal comprovar que no hi hagi cap altre ítem amb el mateix nom, que tingui un tipus vàlid i que no superi el pes màxim. Si se satisfan totes les condicions, es crea l'ítem i es mostra un missatge d'èxit, altrament el programa ha de mostrar un missatge d'error i l'ítem no es crearà.

3. Mostrar tots els jugadors [opció **mostrarJugadors**]

Es mostren les dades de tots els jugadors segons l'ordre en què s'han enregistrat al sistema. Per cada jugador es mostra: el seu nom, el login i el password. Si no hi ha cap jugador enregistrat en el sistema, el programa ha de mostrar un missatge d'error.

4. Mostrar tots els ítems d'equipament [opció **mostrarItems**]

Es mostren les dades de tots els ítems segons l'ordre en què s'han enregistrat al sistema. Per cada ítem es mostra: el seu nom, el tipus i el pes. Si no hi ha cap ítem enregistrat en el sistema, el programa ha de mostrar un missatge d'error.

5. Crear un personatge [opció **crearPersonatge**]

Per a crear un personatge introduïrem el nom, el tipus, el gènere, el pes màxim amb què es pot equipar i el nom del jugador propietari del personatge. El tipus, el gènere i el pes han de complir les especificacions descrites anteriorment i el nom del jugador propietari ha d'existir al sistema. Cal comprovar també que el número de personatges que té associat el jugador no excedeixi del màxim i que el personatge no hagi estat assignat anteriorment a un altre jugador. Si es compleixen totes les condicions, es crearà el personatge, s'assignarà al jugador especificat i es mostrarà un missatge d'èxit. Si alguna de les condicions no es compleix caldrà mostrar un missatge d'error que especifiqui clarament quin error s'ha produït.

6. Consultar personatges d'un jugador [opció **personatgesDeJugador**]

Es mostren les dades del jugador especificant el seu nom d'entrada. Caldrà mostrar tant les dades del jugador com les dades dels personatges que tingui associats. En cas que el nom de jugador donat a l'entrada no existeixi, caldrà mostrar un missatge d'error. Si el jugador no té personatges, també caldrà mostrar el missatge corresponent.

7. Equipar un personatge [opció **equiparPersonatge**]

Per a equipar un personatge introduïrem el nom del personatge i el nom de l'ítem d'equipament. Si el personatge existeix, l'ítem existeix, l'equipament no excedeix del límit establert i el personatge no té equipat ja l'ítem, llavors l'ítem s'associa al personatge. En cas que es produeixi alguna situació anòmla, caldrà mostrar un missatge d'error especificant clarament quin ha estat



el problema.

8. Consultar l'equipament d'un personatge [opció **equipamentDePersonatge**]

Es mostren les dades del personatge especificant el seu nom donat d'entrada. Caldrà mostrar tant les dades del personatge com les dades de tot el seu equipament i el pes total que carrega. En cas que el nom de personatge donat a l'entrada no existeixi, caldrà mostrar un missatge d'error i si el personatge no està equipat amb cap ítem, també caldrà indicar-ho.

9. Sortir del programa [opció **acabar**].

Es garanteix que el fitxer d'entrada sempre contindrà aquesta opció.

Qüestions a tenir en compte:

- Les dades del sistema de jugadors, personatges i ítems només es mantenen durant l'execució del programa, és a dir, no es guarden en cap fitxer extern.
- No hi ha operacions d'esborrats ni modificacions durant l'execució.

2.4 Quarta tasca

La [quarta tasca](#) serà provar l'executable amb els jocs de prova que us facilitem.



3 Exemples d'algunes de les operacions demanades

Nota: Les línies de separació tenen 20 guions si són llargues i 10 si són curtes.

entrada: jocsimple1.ent	sortida: jocsimple1.cor
crearJugador Alex Alex71 4LeX71	Jugador Alex creat correctament Jugador Jordi creat correctament Item Daga creat correctament Item Casc creat correctament Item Capa creat correctament
crearJugador Jordi Jordi80 70Rdi80	Personatge Momo creat correctament per al jugador Alex Personatge Lilith creat correctament per al jugador Alex Personatge Sam creat correctament per al jugador Jordi -----
crearItem Daga Arma 0.3	Tots els jugadors Nom: Alex Login: Alex71 Password: 4LeX71
crearItem Casc Armadura 2	Nom: Jordi Login: Jordi80 Password: 70Rdi80 -----
crearItem Capa Roba 0.5	Tots els Items d'equipament Nom: Daga Tipus: Arma Pes: 0.3
crearPersonatge Momo Guerrer Masculi 30 Alex	Nom: Casc Tipus: Armadura Pes: 2 Nom: Capa Tipus: Roba Pes: 0.5
crearPersonatge Lilith Paladi Femeni 45 Alex	----- Tots els personatges de Nom: Alex Login: Alex71 Password: 4LeX71



crearPersonatge Sam Guerrer Masculi 60 Jordi	Nom: Momo Tipus: Guerrer Genere: Masculi Pes Maxim: 30 ----- Nom: Lilith Tipus: Paladi Genere: Femeni Pes Maxim: 45 -----
mostrarJugadors mostrarItems	----- -----
personatgesDeJugador Alex	Tots els personatges de Nom: Jordi Login: Jordi80 Password: 70Rdi80
personatgesDeJugador Jordi	Nom: Sam Tipus: Guerrer Genere: Masculi Pes Maxim: 60 -----
acabar	



entrada: jocsimple2.ent	sortida: jocsimple2.cor
crearJugador Alex Alex71 4LeX71	Jugador Alex creat correctament Jugador Jordi creat correctament Item Daga creat correctament Item Casc creat correctament Item Capa creat correctament
crearJugador Jordi Jordi80 70Rdi80	Personatge Momo creat correctament per al jugador Alex Personatge Lilith creat correctament per al jugador Alex Personatge Sam creat correctament per al jugador Jordi Personatge Sam equipat amb Capa Personatge Sam equipat amb Casc Personatge Momo equipat amb Casc
crearItem Daga Arma 0.3	----- Equipament de Nom: Sam Tipus: Guerrer
crearItem Casc Armadura 2	Genere: Masculi Pes Maxim: 60 Nom: Capa Tipus: Roba Pes: 0.5
crearItem Capa Roba 0.5	----- Nom: Casc Tipus: Armadura Pes: 2
crearPersonatge Momo Guerrer Masculi 30 Alex	----- Pes total per Sam: 2.5 -----
crearPersonatge Lilith Paladi Femeni 45 Alex	
crearPersonatge Sam Guerrer Masculi 60 Jordi	



equiparPersonatge
Sam
Capa

equiparPersonatge
Sam
Casc

equiparPersonatge
Momo
Casc

equipamentDePersonatge
Sam

acabar