



UNIVERSITÀ
DEGLI STUDI
FIRENZE



Esercitazione

Class exercise

Obiettivo

- Il progetto CLion fornito contiene classi e scheletri di classi relative al gioco in stile Rogue (<https://it.wikipedia.org/wiki/Roguelike>) della scorsa esercitazione.
- Scopo della presente esercitazione è:
 - Completare le classi GameCharacter e Weapon, scrivendo codice di diversi metodi
 - Costruttori, distruttori, operatori, metodi statici
 - Completare una nuova classe con attributo statico

Schema del codice

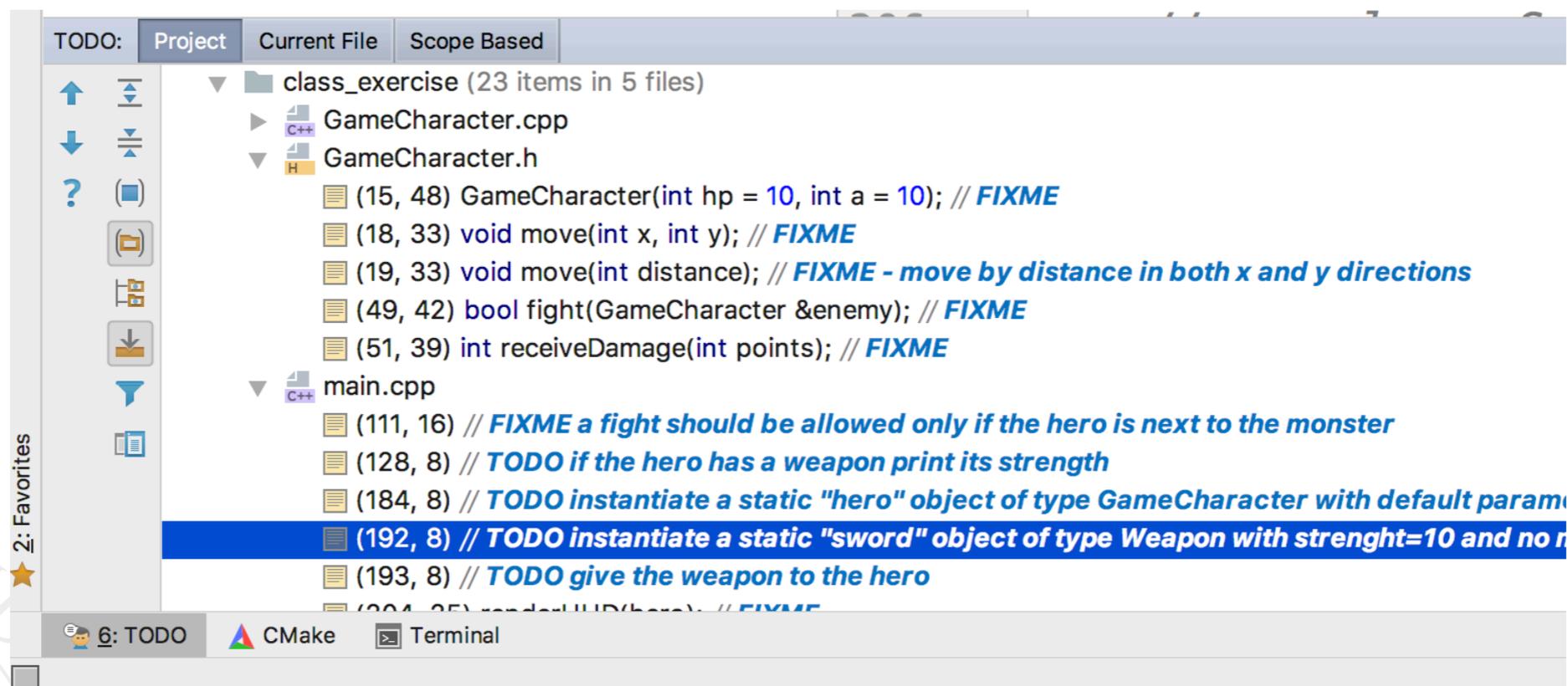
- Il programma è composto da 4 classi:
- Dungeon crea mappe casuali con stanze, corridoi, scale, porte, etc.
- Weapon rappresenta un arma con forza e magia
- GameCharacter rappresenta un personaggio del gioco, ed è composto con Weapon.
- Dice rappresenta un dado

Schema del codice

- Lo schema del codice è lo stesso di quello della volta scorsa
- L'unico cambiamento che faremo è usare il dado per rendere variabile il comportamento del combattimento

Dove modificare il codice

- Le indicazioni precise sul codice da modificare sono fornite come commenti indicati con TODO e FIXME
- Per vedere tutti questi commenti selezionare la finestra TODO di CLion



Dove modificare il codice

The screenshot shows the CLion IDE interface. The main editor displays the code in `main.cpp` with the following content:

```
185 // find a legal start position
186 int startX = 0;
187 int startY = 0;
188 setupCharacterCell(startX, startY, map);
189 hero.setPosX(startX);
190 hero.setPosY(startY);
191 // create a weapon and give it to hero
192 // TODO instantiate a static "sword" object of type Weapon with strenght=10 and
193 // TODO give the weapon to the hero
194 // create an enemy with a low grade armor
195 GameCharacter enemy(20, 2);
196 // find monster position not too far from hero position
197 startX += 5;
198 startY += 3;
199 setupCharacterCell(startX, startY, map);
200 enemy.setPosX(startX);
201 enemy.setPosY(startY);
202
203 // render
204 renderHUD(hero); // FIXME
205 renderGame(map, hero, enemy);
```

The TODO list at the bottom of the IDE shows the following items:

- (15, 48) `GameCharacter(int hp = 10, int a = 10);` // FIXME
- (18, 33) `void move(int x, int y);` // FIXME
- (19, 33) `void move(int distance);` // FIXME - move by distance in both x and y directions
- (49, 42) `bool fight(GameCharacter &enemy);` // FIXME
- (51, 39) `int receiveDamage(int points);` // FIXME
- (111, 16) // FIXME a fight should be allowed only if the hero is next to the monster
- (128, 8) // TODO if the hero has a weapon print its strength
- (184, 8) // TODO instantiate a static "hero" object of type GameCharacter with default parameters
- (192, 8) // TODO instantiate a static "sword" object of type Weapon with strenght=10 and no magic
- (193, 8) // TODO give the weapon to the hero
- (204, 25) `renderHUD(hero);` // FIXME

The status bar at the bottom indicates the current context is `class_exercise [D]`.



Classe Weapon

- Implementare il costruttore di default
- Valutare se usare explicit e delega
- Usare lista di inizializzazione per gli attributi



Classe Dice

- Implementare costruttore
- Aggiungere costante che indica il numero massimo di lanci di dado dopo il quale si re-inizializza il generatore casuale di numeri (RNG)
- Aggiungere attributo statico che conta quanti lanci di dadi sono stati fatti
 - Usare l'attributo per re-inizializzare il generatore casuale di numeri
- Maggiori dettagli sull'uso dei dadi in giochi D&D:
<https://www.redblobgames.com/articles/probability/damage-rolls.html>

Classe GameCharacter

- Implementare il costruttore di default
- Valutare se usare explicit e delega
- Implementare costruttore di copia, operatore di assegnazione e distruttore
- Implementare metodo statico che calcoli la distanza L1 tra due GameCharacter:

$$d_1(p, q) = \| p - q \|_1 = \sum_i | p_i - q_i |$$

- Usare la classe Dice per decidere se si colpisce l'avversario nel combattimento e quanti danni si fanno



Main

- Istanziare oggetti: sword deve essere stanziata sullo heap, così che possa essere distrutta dal distruttore di GameCharacter





Classe Dungeon

- Questa classe non deve essere toccata





Tutte le classi

- Aggiungere la constancy a metodi, variabili, argomenti e attributi ove si ritenga possibile e utile.

