

# Exercici 2

Cal aplicar el patró STRATEGY per agrupar algorismes relacionats, i que la classe Character pugui utilitzar i inclòs intercanviar durant l'execució.

Amb aquest patró evitarem l'ús repetit de condicionals i switch. Com es pot veure al codi inicial, hi ha dos mètodes a attacking i defending, per tant, aplicarem dues vegades el patró amb les interfícies IAttackStrategy i IDefenseStrategy. La classe Character tindrà dos atributs nous per guardar les estratègies escollides, com a resultat s'optimitza aquesta classe, donat que les estratègies concretes només s'assignen a la creació del personatge. El personatge Knight és singular perquè pot canviar la defensa.

Donat el codi inicial, es decideix que les estratègies concretes siguin singleton, però es podria donar el cas que en el futur, per exemple, tinguem dues instàncies de la

