

Feladat – NagyZH – FONTOS INFÓK

- Alkalmazni kell a megoldás során a tanult objektum-orientáltsági elveket.
- A bemeneti fájlokat (**in-arena.json**, **in-actions.txt**) teszteléskor a build könyvtárba kell másolni.
- Feltehetjük, hogy a bemenő/teszt adatok helyesek, érvényesek (kivéve, ha a feladat mást nem állít).
- A megoldás teljes forráskódját egyetlen ZIP fájlba csomagolva kell feltölteni.

A feladat leírása

A *main*-ben és ebben a leírásban szereplő osztályok és metódusok igény szerint **átnevezhetők**, az itteni nevek csak példák.

A feladatban játékos karakterek menedzselése a cél, akik egy arénában harcolnak egymással. Mindenkinek van egy szöveges azonosítója, életereje (HP) és maximális életereje. Kezdetben az életerő mindig megegyezik a maximális életerővel. Minden szereplő három karakterosztály valamelyikébe tartozik: katona, mágus vagy pap. A katonának van ereje (power), és páncélja (armor). A mágusnak és a papnak van varázsereje (magic). Az említett tulajdonságok mindegyike egész és egyike sem negatív.

Legyen egy megfelelő osztályszerkezet, amiben tárolni tudjuk a különböző szereplőket. Az **Arena** osztály *load* metódusa legyen képes beolvasni egyetlen JSON fájlból mindenkit. A "**CharacterClass**" kulcs mondja meg, hogy ki melyik karakterosztályba tartozik. A *print* metódus azonosító szerint keressen meg egy szereplőt és írja ki minden adatát. A *printAll* metódus írja ki az összes szereplő minden adatát.

Tudjanak a szereplők akciókat végrehajtani egymáson. Ez két, függetlenül pontozott feladatrészből áll.

Egyrészt, az **Arena** osztályban legyen egy *performAction* metódus, ami megkapja az akció végrehajtóját és a célpontot. Nem kell kiírni semmit. Az okozott hatás a két fél karakterosztályától függ az alábbiak szerint.

- A katona megtámadja a célpontját, akinek az életereje annnyival csökken, amennyi a katona ereje. Ha viszont van a célpontnak páncélja, akkor az semlegesíti az okozott sérülésből annyit, amennyi a páncél értéke. A célpont ezzel együtt is, mindenképp sérül legalább 1-et.
- A mágus is megtámadja a célpontját, akinek az életereje annnyival csökken, amennyi a mágus varázsereje. A védekező páncéljának itt nincs hatása.
- A pap gyógyítja a célpontját, akinek az életereje annnyival nő, amennyi a pap varázsereje.
- Soha nem csökkenhet egy szereplő életereje 0 alá, és nem nőhet a maximális életereje fölé. Ha ez történné, akkor az életerő nulla, illetve maximális marad.
- Ha egy szereplő életereje 0-ra csökken, akkor elesik. Innentől őt nem lehet gyógyítani és nem tud akciókat végrehajtani (vagyis azoknak nincs hatása).

Másrészt, lehessen ilyen akciók sorozatát a programba beolvasni és tárolni egy **ActionList** osztályban. Ez konstruktorban egy szöveges fájlt vár, innen beolvassa és eltárolja a végrehajtó-célpont párokat. Az azonosítók nem tartalmazznak whitespace-t. Legyen egy *printSome* metódus, ami kiír *néhányat* ezekből (nem mindet, az sok is lehet). Az **Arena** osztályban legyen egy *performList* metódus, ami a kapott **ActionList**-ben tárolt akciókat sorban végrehajtja.

Pontozás

- | | |
|--|-----------|
| 1. A karakterek osztályai, megfelelő adatszerkezet | (8 pont) |
| 2. Arena , beolvasás JSON fájlból | (6 pont) |
| 3. Arena , <i>print</i> és <i>printAll</i> metódusok | (6 pont) |
| 4. <i>performAction</i> metódus, a különféle szabályok kezelése | (12 pont) |
| 5. ActionList osztály, adatszerkezet, beolvasás szövegfájlból | (4 pont) |
| 6. <i>printSome</i> metódus | (2 pont) |
| 7. Az Arena -ban végrehajtjuk a teljes ActionList -et | (2 pont) |

Összesen: **40 pont.**