

Less Fun Version of Warcraft 3

Bevezető

A Warcraft 3 egy olyan stratégiai játék, amelyben az ellentétes csapatok, miután létrehozták saját bázisukat, csatával döntenek el, hogy ki legyen a végső nyertes. A feladat egy hasonló játék szimulálása, leegyszerűsítve. Az alapfeladatban egymással háborúzni nem kell, csupán létrehozni saját bázisunkat, a kiegészítésben pedig létrehozni a katonákat, majd szimulálni a mindent eldöntő csatát két bázis között.

A **Building**, **Resources**, **Simulation**, **Unit**, **UnitType** osztályokat ne módosítsuk a feladatok teljesítése során.

Alapfeladat – 5 pont

A **Simulation** osztály a program belépési pontja, itt hozunk létre egy **Base** objektumot. A **Base** osztály konstruktorának egy paramétere van; a bázis neve.

A bázis feladata, hogy létrehozzon bizonyos épületeket, illetve munkásokat. Mind a munkásoknak, mind az épületeknek vannak közös tulajdonságai; mennyi aranyba kerül létrehozni (**goldCost**), mennyi fába kerül létrehozni (**woodCost**), mennyi ételbe kerül fenntartani (**foodCost**, ez az épületeknél 0), melyik bázishoz tartozik (**owner**), mi a típusa (**unitType**), és mennyi időbe telik elkészíteni (**buildTime**). Ezen információk tárolására hoztuk létre a **Unit** absztrakt osztályt, ami számon tartja melyik bázishoz tartozik az adott egység (**owner**), illetve hogy milyen típusú az adott egység (**unitType**). Az egységek típusát egy Enummal határozzuk meg, amelyben tároljuk a **goldCost**, **woodCost**, **foodCost**, **buildTime** attribútumokat is, ennek a neve **UnitType**.

A **UnitType** lehetséges értékei:

Name	Gold Cost	Wood Cost	Food Cost	Build Time (ms)
FARM	80	20	0	2000
LUMBERMILL	120	0	0	4000
BLACKSMITH	140	60	0	5000
PEASANT	75	0	1	1000

A **Resources** osztály segít számon tartani, hogy egy bázis milyen nyersanyagokkal rendelkezik (**gold**, **wood**), illetve mennyi embert tud ellátni (**capacity**, **capacityLimit**). A **capacity** mondja meg, hogy a jelenleg létrehozott egységek ellátására mennyi élelmiszer szükséges, a **capacityLimit** pedig azt, hogy maximum mennyit tudunk ellátni. **A capacity nem haladhatja meg a capacityLimit-et!** A **capacityLimit** kezdő értéke legyen 5.

A **Peasant** osztály a dolgozókat fogja szimbolizálni. Ez a **Unit** osztályból származik le, és a következő funkciókkal rendelkezik, melyeket implementálni kell a feladat teljesítéséhez:

- **startMining**: Elkezd aranyat termelni. Minden 100ms elteltével termel 10 aranyat.

- **startCuttingWood:** Elkezd fát vágni. Minden 100ms elteltével termel 10 fát. **A dolgozó nem tud egyszerre fát is vágni is aranyat is termelni.**
- **tryBuilding:** Megpróbál felépíteni egy épületet, amennyiben van elegendő erőforrása annak felépítésére, és éppen nem aranyat vagy fát termel. Ez visszafele is igaz – amíg építkezik, addig nem tud aranyat vagy fát termelni.

A **Building** osztály szimbolizálja az épületeket. A *Lumbermill* és *Blacksmith* nem befolyásolja a játék menetét. Minden *Farm* 10-zel növeli az élelmiszer ellátási kapacitást a **Resources** osztályban.

A **Base** osztály konstruktorában hozzunk létre 5 **Peasant**-et, amiből 3 elkezd aranyat termelni, egy fát vágni, egy pedig nem csinál semmit kezdetben. Az osztály rendelkezik egy **startPreparation** metódussal. Ennek metódusnak a feladata, hogy elérjük az alábbi állapotot

- Legyen 10 **Peasant**, amiből 5-en aranyat termelnek, 2-en fát vágnak. Figyeljünk, hogy ne legyen több **Peasant**, mint amennyi ellátására elegendő élelmiszerünk van. Egyszerre csak egy **Peasant**-ot lehessen képezni.
- Rendelkezzünk 3 *Farm*-mal.
- Rendelkezzünk egy *Blacksmith*-tel.
- Rendelkezzünk egy *Lumbermill*-l.

Miután ezt elértük, a dolgozók hagyják abba a nyersanyagtermelést. Amint van nyersanyag és kapacitás megépíteni valamit, vagy létrehozni egy új **Peasant**-ot, egyből kezdjük is azt el.

Kiegészítés – 10 pont

Az új **Personnel** osztály a **Unit**-ből származik le. Ez az osztály az új őse a **Peasant** osztálynak, illetve egy új **Footman** osztálynak. Minden **Personnel** rendelkezzen életpontokkal (**health**), minimum és maximum támadási értékkel (**attackMin**, **attackMax**), illetve egy **Personnel** típusú mezővel, ami az aktuális ellenfelét írja le (**opponent**). Implementáljuk az alábbi metódusokat:

- **startWar:** Paraméterként megkapja az ellenséges hadsereget (**List<Personnel>**), és egy véletlenszerűen kiválasztott tagját addig támadja, amíg annak, vagy saját magának el nem fogy az élete. Amennyiben ellenfele élete fogyott el, addig keres új ellenfelet, amíg van még valaki az ellenséges hadseregben. Minden támadás között teljen el 100-200ms, illetve véletlenszerűen sebezzen ellenfelébe a megadott maximum és minimum támadási értékeken belül. **Megjegyzés:** Nem kell figyelni rá, hogy a katona, akit megtámadtak, azt támadja vissza, aki éppen őt támadja. Azaz elképzelhető, hogy A megtámadja B-t, B pedig C-t.
- **loseHealth:** Életpontok vesztese egy támadást követően. Ha nullára, vagy alá csökken, vegyük ki hadseregből, illetve a **Resources**-ban állítsuk be a kapacitást ennek megfelelően.

Personnel type	Health	Attack Min	Attack max
PEASANT	220	5	6
FOOTMAN	420	12	15

A **Base startPreparation** metódusa megváltozik úgy, hogy most egy *Barracks* típusú épületet is létre kell hozni, illetve 10 **Footman**-t. Ahhoz, hogy **Footman** típusú egységeket tudjunk gyártani, kell, hogy legyen legalább 1 *Barracks*, egyszerre csak egy **Footman**-t lehessen kiképezni.

Name	Gold Cost	Wood Cost	Food Cost	Build Time (ms)
BARRACKS	160	60	0	6000
FOOTMAN	135	0	2	1500

Továbbá a **startPreparation** befejeztével meghívjuk meg az **assembleArmy** metódust, ami létrehozza a bázis hadseregét – Ennek az implementálását be kell fejezni. A hadseregbe kerüljenek be a **Footman** típusú katonák, illetve a **Peasant** dolgozók is, miután befejeztek minden építkezési munkát és nyersanyagtermelést.

Létrehoztunk egy új, ellenséges bázist is. Mindkét bázis működjön az eddig leírtak szerint, majd ha készen vannak az **assembleArmy** feladataival is, meghívják a **goToWar** metódust, melynek implementálását be kell fejezni. A metódus csak akkor kezdődhet el, ha már mindkét bázisnál megtörtént az **assembleArmy**. Ebben a metódusban indítsuk el a támadást a saját összes katonánkkal az ellenfél serege ellen. Mindegyik katona külön szálon harcoljon, egymásra ne várjanak soha. Ha vége a háborúnak (tehát vagy a saját vagy az ellenfél csapata megsemmisült), írjuk ki, hogy az adott bázis nyert-e, vagy veszített.