Tóth Zalán

(CZ72YM)

Feladat leírás:

Egy bolygón különböző fajtájú növények élnek, minden növény tápanyagot használ. Ha egy növény tápanyaga elfogy (a mennyiség 0 lesz), a növény elpusztul. A bolygón három fajta sugárzást különböztetünk meg: alfa sugárzás, delta sugárzás, nincs sugárzás. A sugárzásra a különböző fajtájú növények eltérő módon reagálnak. A reakció tartalmazza a tápanyag változását, illetve a következő napi sugárzás befolyásolását. A másnapi sugárzás alakulása: ha az alfa sugárzásra beérkezett igények összege legalább hárommal meghaladja a delta sugárzás igényeinek összegét, akkor alfa sugárzás lesz; ha a delta sugárzásra igaz ugyanez, akkor delta sugárzás lesz; ha a két igény közti eltérés háromnál kisebb, akkor nincs sugárzás. Az első nap sugárzás nélküli.

***Szimuláljuk a növények viselkedését és minden lépésben írjuk ki az összes növényt a rájuk jellemző tulajdonságokkal, valamint az aktuális sugárzást!***

Minden növény jellemzői: az egyedi neve (String), a rendelkezésre álló tápanyag mennyisége (egész), hogy él-e (logikai). A szimulációban részt vevő növények fajtái a következők: puffancs, deltafa, parabokor. A következőkben megadjuk, hogy miként reagálnak a különböző sugárzásokra. Először a tápanyag változik, és ha a növény ezután él, akkor befolyásolhatja a sugárzást.

Puffancs: Alfa sugárzás hatására a tápanyag mennyisége kettővel nő, sugárzás mentes napon a tápanyag eggyel csökken, delta sugárzás esetén a tápanyag kettővel csökken. Minden esetben úgy befolyásolja a másnapi sugárzást, hogy 10-tápanyag értékben növeli az alfa sugárzás bekövetkezését. Ez a fajta akkor is elpusztul, ha a tápanyag mennyisége 10 fölé emelkedik.

Deltafa: Alfa sugárzás hatására a tápanyag mennyisége hárommal csökken, sugárzás nélküli napon a tápanyag eggyel csökken, delta sugárzás hatására a tápanyag néggyel nő. Ha a tápanyag mennyisége 5-nél kisebb, akkor 4 értékben növeli a delta sugárzás bekövetkezését, ha 5 és 10 közé esik, akkor 1 értékben növeli a delta sugárzás bekövetkezését, ha 10-nél több, akkor nem befolyásolja a másnapi sugárzást.

Parabokor: Akár alfa, akár delta sugárzás hatására a tápanyag mennyisége eggyel nő. Sugárzásnélküli napon a tápanyag eggyel csökken. A másnapi sugárzást nem befolyásolja.

A program egy szövegfájlból olvassa be a szimuláció adatait! Az első sorban a növények száma szerepel. A következő sorok tartalmazzák a növények adatait szóközökkel elválasztva: a növény nevét, a fajtáját és a kezdetben rendelkezésére álló tápanyag mennyiségét. A fajtát egy karakter azonosít: a – puffancs, d – deltafa, p – parabokor. A növényeket leíró részt követő sorban a szimuláció napjainak száma adott egész számként. A program kérje be a fájl nevét, majd jelenítse meg a túlélők nevét! (Feltehetjük, hogy a fájl formátuma helyes.) Egy lehetséges bemenet:

4

Falánk a 7

Sudár d 5

Köpcös p 4

Nyúlánk d 3

10

# Fejlesztői eszközök:

* IntelliJ 2020.3
* Java 15 (preview tartalmak is)
* Gradle 6.7
* Lombok 1.18
* Junit 5.6
* Mockito 3.7

Fordítás: *gradle build*   
Futtatás: *java -jar --enable-preview vegetation-simulation-1.0.0.jar*

# Megoldási terv:

1, Objektum modelek elkésztése.  
A program három növény típust kezel, melyek több közös tulajdonsággal rendelkeznek (név, tápanyag, életben van-e), így egy közös ősosztály létrehozása szükséges.   
  
*Plan(name: String,nutrient: Int, alive: Bool)*  
  
Az ősosztály tartalmazza a közös funkcionalitást is:

* Tápanyag vizsgálata, életben van jelző beállítása
* Sugárzás hatása adott napon

Mivel az adott napi sugárzás hatás kiszámítási módjának folyamatlépései megegyeznek, az ősosztály tartalmazza ezt a funkcionalitást. Az eltéréseket absztrakt metódusok hívásával kezeli, melyeket az alosztályok implementálnak.  
  
Ebből a szülő osztályból származnak a konkrét osztályok (*Puffancs, Deltafa, Parabokor*)  
  
Szükséges a sugárzás típusainak kezelése. Három adott típusú sugárzással dolgozik a program, melyet egy enumerációs osztályban ír le.  
  
*Radiation {ALPHA, DELTA, NEUTRAL}*

Az egyes növények következő napi sugárzásra gyakorolt hatását két tulajdonság ír le. Mely sugárzás esélyét növeli, illetve mennyivel.   
  
*Effect(radiation: Radiation, quantity: int)*

A szimuláció futtatása előtt fájlból olvassuk be a szükséges adatokat. Ezen adatok tárolására szintén létre kell hozni egy osztályt, mely tartalmazza a szimulálandó napok számát, valamint a szimulációban résztvevő növényeket.

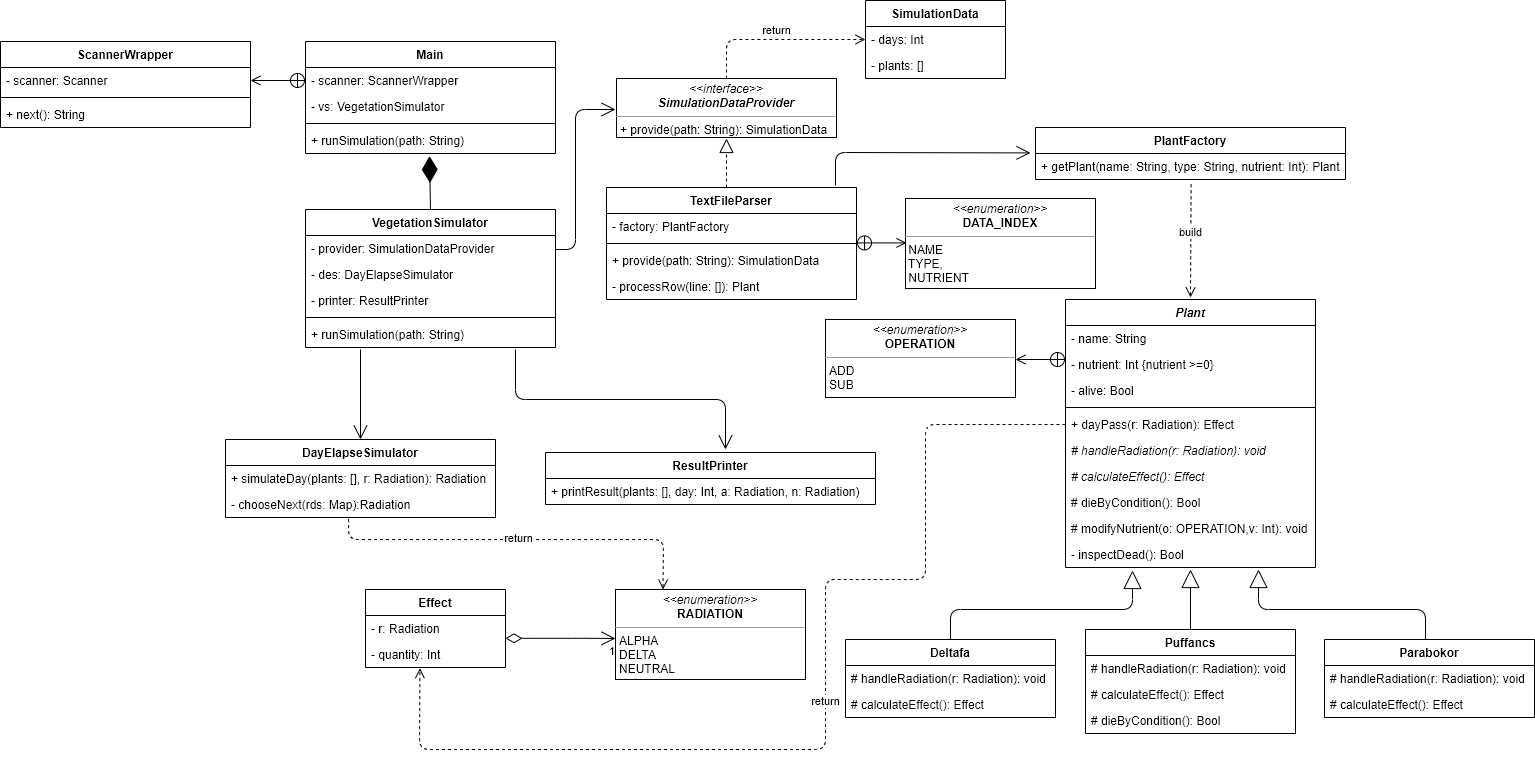
*SimulationData(days: Int, plants: [])*

## 2, Szimuláció megvalósítása

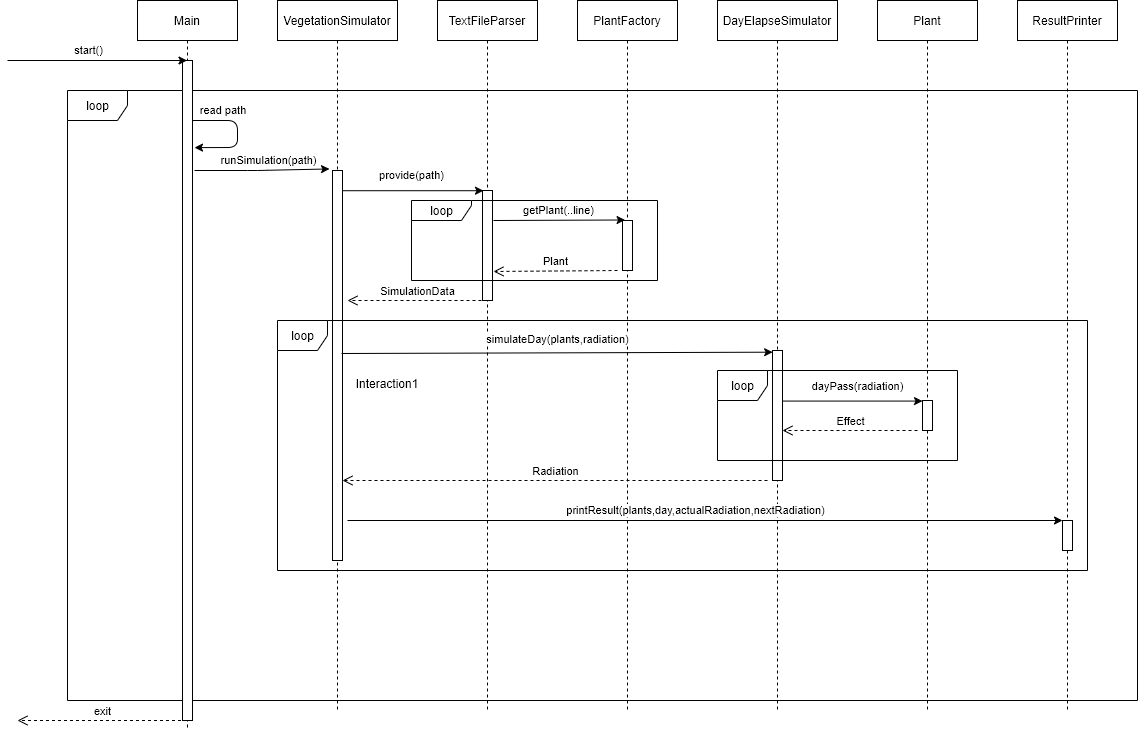
A szimuláció több lépésre bontható.

1. Interakció a felhasználóval, fájl elérési útvonalának bekérése.
2. Fájl beolvasása, növény példányok létrehozása.
3. Napszámszor végrehajtva:
   * 1. Minden növényen az adott napi sugárzás hatásának végrehajtása.
     2. A növények sugárzásra gyakorolt hatásainak összesítése.
     3. Az adott nap eredményének kiírása.

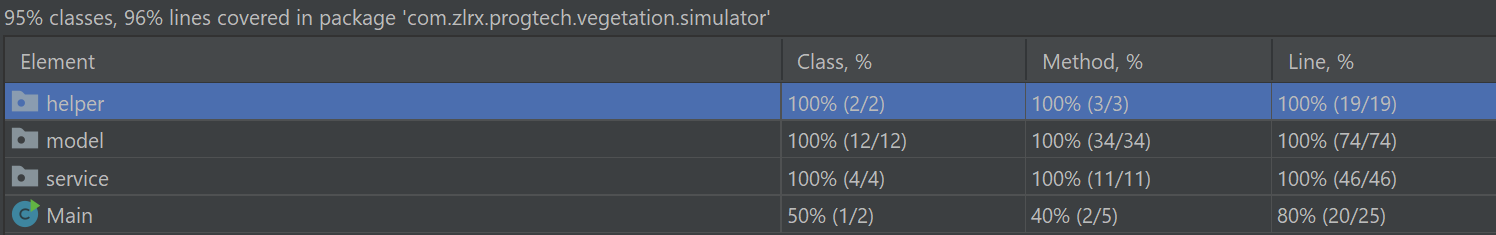
A fenti lépések alapján a funkcionalitás könnyen részeegységekre bontható.   
A felhasználóval történő interakció a *Main* osztályból történik. A *VegetationSimulator* vezérli a szimulációs folyamatot. A *TextFileParser* osztály az adatok beolvasásáért felelős, mely alapján a *PlantFactory* segítségével legyártja a szükséges növény (*Plant*) példányokat.  
Az egyes napokon végrehajtandó számításokat a *DayElapseSimulator* osztály kezeli, mely visszaadja a következő napi sugárzást (*Radiation*). A *VegetationSimulator* a következő ciklus-iterációban (napon) ezzel az értékkel hívja a *DayElapseSimulator* kalkulációs metódusát. Az eredményeket minden ciklus-iterációban a *ResultPrinter* osztály segítségével írja a képernyőre.  
  
Osztály diagram:



## Szekvencia diagram:



Tesztelési terv:

Cél az egyes osztályok fő funkcióinak és határeseteinek minél szélesebb körű lefedése egységtesztekkel. Valamint az IO műveletek megvizsgálása integrációs tesztekkel.  
  


|  |  |  |  |
| --- | --- | --- | --- |
| Tesztelt metódus | Leírás | Bemenet | Elvárt eredmény |
| PlantFactory.dayPass (PlantFactoryTest) | Puffancs létrehozás | ’a’ típus | Puffancs objektumot tartalmazó Optional |
| Deltafa létrehozás | ’d’ típus | Deltafa objektumot tartalmazó Optional |
| Parabokor létrehozás | ’p’ típus | Parabokor objektumot tartalmazó Optional |
| Üres eredmény nem ismert típus esetén | ’x’ típus | Optional.empty() |
| Deltafa.dayPass (DeltafaTest) | Alfa sugárzás hárommal csökkenti a tápanyag mennyiséget | deltafa 7 tápanyaggal alfa sugárzás | deltafa 4 tápanyaggal |
| Delta sugárzás néggyel növeli a tápanyag mennyiségét | deltafa 1 tápanyaggal delta sugárzás | deltafa 5 tápanyaggal |
| Sugárzásmentes napon eggyel csökken a tápanyag mennyisége | deltafa 12 tápanyaggal semleges sugárzás | deltafa 11 tápanyaggal |
| Elpusztul ha a tápanyag nullára csökken | deltafa 2 tápanyaggal alfa sugárzás | deltafa 0 tápanyaggal életben jelző hamis |
| Öt tápanyag alatt +4 egységgel növeli a delta sugárzás esélyét | deltafa 7 tápanyaggal alfa sugárzás | deltafa 4 tápanyaggal hatás 4 delta sugárzás |
| Öt és tíz között +1 egységgel növeli az delta sugárzás esélyét | deltafa 13 tápanyaggal alfa sugárzás | deltafa 10 tápanyaggal hatás 1 delta sugárzás |
| Tíz felett nem növeli a sugárzás esélyét | deltafa 14 tápanyaggal alfa sugárzás | deltafa 11 tápanyaggal hatás 0 sugárzással |
| Parabokor.dayPass (ParabokorTest) | Alfa sugárzás eggyel növeli a tápanyag mennyiséget | parabokor 5 tápanyaggal alfa sugárzás | parabokor 6 tápanyaggal hatás 0 sugárzás |
| Delta sugárzás eggyel növeli a tápanyag mennyiségét | parabokor 5 tápanyaggal delta sugárzás | parabokor 6 tápanyaggal hatás 0 sugárzás |
| Sugárzásmentes napon eggyel csökken a tápanyag mennyisége | parabokor 5 tápanyaggal semleges sugárzás | parabokor 6 tápanyaggal hatás 0 sugárzás |
| Elpusztul ha a tápanyag nullára csökken | parabokor 1 tápanyaggal semleges sugárzás | parabokor 0 tápanyaggal életben jelző hamis hatás 0 sugárzás |
| Puffancs.dayPass Plants constuctor (PuffancsTest) | Elpusztult puffancs nincs hatással a sugárzásra | elpusztult puffancs | hatás 0 sugárzás |
| Nullánál kisebb tápanyaggal elpusztulva jön létre | -1 tápanyag konstruktorba | puffancs 0 tápanyaggal életben jelző hamis |
| Alfa sugárzás kettővel növeli a tápanyag mennyiséget | puffancs 8 tápanyaggal alfa sugázás | puffancs 10 tápanyaggal hatás 0 alfa |
| Delta sugárzás kettővel csökkenti a tápanyag mennyiségét | puffancs 3 tápanyaggal delta sugárzás | puffancs 1 tápanyaggal hatás 9 alfa |
| Sugárzásmentes napon eggyel csökken a tápanyag mennyisége | puffancs 5 tápanyaggal semleges sugárzás | puffancs 4 tápanyaggal hatás 6 alfa |
| Elpusztul ha a tápanyag nullára csökken | puffancs 1 tápanyaggal delta sugárzás | puffancs 0 tápanyaggal életben jelző hamis hatás - sugárzás |
| Elpusztul ha a tápanyag tiz fölé emelkedik | puffancs 8 tápanyaggal alfa sugárzás | puffancs 11 tápanyaggal életben jelző hamis hatás - sugárzás |
| DayElapseSimulator  .simulateDay  (DayElapseSimulatorTest) | Ha minden növény elpusztult, semleges sugárzást várunk | elpusztult puffancs, deltafa, parabokor | semleges sugárzás |
| Ha az alfa hatások összessége legalább hárommal nagyobb a delta hatások összességénél akkor alfa sugárzás keletkezik | plants objektumok: puffancs 1 deltafák 8, 8, 13, 11 alfa sugárzás | alfa sugárzás |
| Ha az delta hatások összessége legalább hárommal nagyobb a alfa hatások összességénél akkor delta sugárzás keletkezik | plants objektumok: puffancs 5 deltafák 6, 6, 5, 11 alfa sugárzás | delta sugárzás |
| Ha delta összege kevesebb mint hárommal több alfa hatás összegénél, akkor semleges sugárzás keletkezik | plants objektumok: puffancs 5 deltafák 6, 8 alfa sugárzás | semleges sugárzás |
| Ha alfa hatás összege kevesebb mint hárommal több delta hatás összegénél, akkor semleges sugárzás keletkezik | plants objektumok: puffancs 4 deltafák 8, 9 alfa sugárzás | semleges sugárzás |
| TextFileParser .provide  (TextFileParserIntTest) | Nem létező fájl esetén nem kap adatot | hibás fájl útvonal | Optional.empty() |
| Létező file esetén felolvassa a filet és létrehozza a szimulációs adatokat | resource mappában található fájl útvonala | Optionalba csomagol szimulációs adatok a fájlnak megfelelő adatokkal |
| VegetationSimulator .runSimulation (VegetationSimulatorTest) | Ha nincs adat, egyszer sem hívódik meg a kiírás és a kalkulációs metódusok | nem létező file útvonal (mock alter) | nincs interakció az elvárt metódusokkal |
| Megfelelő bemenettel meghívódnak a kalkulációs metódusok és jó végeredményt adnak | Puffancs 10 Parabokor 3 Deltafák 7, 5, 5, 7 | Megfelelő számú interakció Helyes végeredmény |
| Main.start (MainE2ETest) | A folyamat elindul, feldolgozza a fájlt és exit parancsra kilép 1 másodperces időlimiten belül | - | lsd leírás |

# Teszt riportok:

