## 1. Parancsok listája – Felülvizsgálat

***A 11. számú use-case kapcsán***

**CREATE\_MUSHROOMBODY** MushroomBody\_Name Tecton\_Name

**Leírás:** A gombatest létrejön egy tektonon (céltekton).

(Megjegyzések:

* a gombatestek elnevezési konvenciója: mb[1-től kezdődő, folytatólagos számozás];
* a létrejött gombatest hátralévő spórakilövéseinek száma alapértelmezetten 3;
* a létrejött gombatest spóralistája alapértelmezetten üres, azzal, hogy a játék elindítása utáni első kör elején már termelődik spóra a gombatestben.)

**Opciók:** Az első paraméter meghatározza a létrehozandó gombatestet, a második a céltektont.

***A 12. számú use-case kapcsán***

**GROW\_MUSHROOMBODY** MushroomBody\_Name Tecton\_Name

**Leírás:** A gombatest létrejön és rákerül egy tektonra (céltekton).

(Megjegyzések:

* a gombatestek elnevezési konvenciója: mb[a CREATE\_MUSHROOMBODY paranccsal létrehozott gombatestek számozását folytató, ennek hiányában 1-től kezdődő, folytatólagos számozás];
* a létrejött gombatest hátralévő spórakilövéseinek száma 3;
* a létrejött gombatest spóralistája üres, azzal, hogy a gombatest a létrehozatalára vonatkozó parancs kiadását követően azonnal létrejön és az ezt követő első kör elején már termelődik benne spóra.)

**Opciók:** Az első paraméter meghatározza a létrejövő gombatestet, a második a céltektont.

***A 13. számú use-case kapcsán***

**PUT\_SPORE** Spore\_TypeSpore\_Name Tecton\_Name

**Leírás:** Egy adott típusú spóra rákerül egy tektonra (céltekton).

(Megjegyzés:

* a spórák elnevezési konvenciója: [a spóra típusára utaló elnevezés, azaz: StunSpore esetén stuns; PreventCutSpore esetén prevents; SlownessSpore esetén slows; SpeedSpore esetén speeds; SplitSpore esetén splits][1-től kezdődő, folytatólagos számozás – minden spóratípus esetén külön-külön].)

**Opciók:** Az első paraméter meghatározza a spóra típusát, a második a nevét, a harmadik a céltektont.

***A 14. számú use-case kapcsán***

**EJECT\_SPORES** MushroomBody\_Name Tecton\_Name

**Leírás:** A kiválasztott gombatest valamennyi spórája rákerül egy tektonra (céltekton).

(Megjegyzések:

* a gombatestben körönként automatikusan termelődő spórák elnevezési konvenciója: [gombatest neve]-[a spóra típusára utaló elnevezés, azaz: StunSpore esetén stuns; PreventCutSpore esetén prevents; SlownessSpore esetén slows; SpeedSpore esetén speeds; SplitSpore esetén splits][1-től kezdődő, folytatólagos számozás – minden spóratípus esetén külön-külön];
* a tesztesetekben a gombatestben körönként automatikusan termelődő spóra SpeedSpore típusú.)

**Opciók:** Az első paraméter meghatározza a gombatestet, a második a céltektont.

***A 15. számú use-case kapcsán***

**DEACTIVATE** MushroomBody\_Name

**Leírás:** A kiválasztott gombatest inaktívvá válik.

**Opciók:** A paraméter meghatározza a gombatestet.

***Új parancs***

**ADD\_SPORE** Spore\_TypeSpore\_Name MushroomBody\_Name

**Leírás:** A rendszer a kiválasztott gombatesthez meghatározott típusú spórát rendel.

(Megjegyzés:

* a spórák elnevezési konvenciója: [a spóra típusára utaló elnevezés, azaz: StunSpore esetén stuns; PreventCutSpore esetén prevents; SlownessSpore esetén slows; SpeedSpore esetén speeds; SplitSpore esetén splits][a PUT\_SPORE paranccsal létrehozott spórák számozását folytató, ennek hiányában 1-től kezdődő, folytatólagos számozás].)

**Opciók:** Az első paraméter meghatározza a spóra típusát, a második a spóra nevét, harmadik a gombatestet.

***Új parancs***

**SET\_REMAININGEJECTS** MushroomBody\_Name RemainingEjects\_Count

**Leírás:** A rendszer a kiválasztott gombatesthez meghatározott számú, hátralévő spórakilövést rendel.

**Opciók:** Az első paraméter meghatározza a gombatestet, a második a hátralévő spórakilövések számát.

## 2. A tesztek részletes tervei, leírásuk a teszt nyelvén

[A tesztek részletes tervei alatt meg kell adni azokat a bemeneti adatsorozatokat, amelyekkel a program működése ellenőrizhető. Minden bemenő adatsorozathoz definiálni kell, hogy az adatsorozat végrehajtásától a program mely részeinek, funkcióinak ellenőrzését várjuk és konkrétan milyen eredményekre számítunk, ezek az eredmények hogyan vethetők össze a bemenetekkel. A tesztek leírásakor az előző dokumentumban (proto koncepciója) megadott szintakszist kell használni.]

### TSZ/1. számú teszteset – Gombatest sikeres növesztése FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton) gombafonál által

* **Leírás**

Gombafonál sikeresen növeszt gombatestet olyan FertileTectonra, amely nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton.

(Megjegyzés: SustainingTecton, MultiLayeredTecton és AridTecton esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikeres növesztése az ehhez szükséges feltételekkel: a FertileTectonon mint céltektonon van legalább 3 db spóra és nincs gombatest
  + a létrejött gombatest spóráinak meghatározása
  + a létrejött gombatest megmaradt spórakilövései számának beállítása
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

PUT\_SPORE SpeedSpore speeds1 ft1

PUT\_SPORE SpeedSpore speeds2 ft1

PUT\_SPORE SpeedSpore speeds3 ft1

ADD\_PLAYER Mycologist mycologist1

CREATE\_MYCELIUM Mycelium m1

ADD\_MYCELIUM\_TO\_TECTON m1 ft1

START\_GAME

GROW\_MUSHROOMBODY mb1 ft1

ENDTURN

STATE ft1

STATE m1

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

}

myceliumCapacity int = 1

spores Queue<Spore> = {

speeds1

speeds2

speeds3

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

m1

}

occupants List<Insect> = {

}

m1: Mycelium

growing boolean = false

location Tecton = ft1

growTimer int = 0

mb1: MushroomBody

remainingEjects int = 3

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds1

}

### TSZ/2. számú teszteset – Gombatest spórahiány miatti sikertelen növesztése FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton) gombafonál által

* **Leírás**

Gombafonál sikertelenül kísérel meg gombatestet létrehozni olyan FertileTectonra, amely nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton, és amelyen nem található elegendő spóra.

(Megjegyzés: SustainingTecton, MultiLayeredTecton és AridTecton esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikertelen növesztése a következő feltételek fennállása mellett: a FertileTectonon mint céltektonon nincs legalább 3 db spóra és nincs gombatest
  + az objektumok állapotában nem következik be változás
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_MYCELIUM Mycelium m1

PUT\_SPORE SpeedSpore speeds1 ft1

PUT\_SPORE SpeedSpore speeds2 ft1

ADD\_PLAYER Mycologist mycologist1

ADD\_MYCELIUM\_TO\_TECTON m1 ft1

START\_GAME

GROW\_MUSHROOMBODY mb1 ft1

ENDTURN

STATE ft1

STATE m1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

}

myceliumCapacity int = 1

spores Queue<Spore> = {

speeds1

speeds2

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m1

}

occupants List<Insect> = {

}

m1: Mycelium

growing boolean = false

location Tecton = ft1

growTimer int = 0

### TSZ/3. számú teszteset – Gombatest sikertelen növesztése gombafonál által olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amelyen már van gombatest

* **Leírás**

Gombatest sikertelen növesztése gombafonál által olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amelyen már van gombatest.

(Megjegyzés: SustainingTecton, MultiLayeredTecton és AridTecton esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikertelen növesztése a következő feltételek fennállása mellett: a FertileTectonon mint céltektonon van gombatest és legalább 3 db spóra
  + az objektumok állapotában nem következik be változás
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

PUT\_SPORE SpeedSpore speeds1 ft1

PUT\_SPORE SpeedSpore speeds2 ft1

PUT\_SPORE SpeedSpore speeds3 ft1

ADD\_PLAYER Mycologist mycologist1

CREATE\_MYCELIUM Mycelium m1

ADD\_MYCELIUM\_TO\_TECTON m1 ft1

CREATE\_MUSHROOMBODY mb1 ft1

START\_GAME

GROW\_MUSHROOMBODY mb2 ft1

ENDTURN

STATE ft1

STATE m1

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

}

myceliumCapacity int = 1

spores Queue<Spore> = {

speeds1

speeds2

speeds3

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

m1

}

occupants List<Insect> = {

}

m1: Mycelium

growing boolean = false

location Tecton = ft1

growTimer int = 0

mb1: MushroomBody

remainingEjects int = 3

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds1

mb1-speeds2

}

### TSZ/4. számú teszteset – Gombatest sikertelen növesztése gombafonál által SemiFertileTectonra

* **Leírás**

Gombafonál sikertelenül kísérel meg gombatestet létrehozni SemiFertileTectonra, amelyen van legalább 3db spóra (és nincs, mert nem is lehet rajta gombatest).

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikertelen növesztése a következő feltételek fennállása mellett: a SemiFertileTectonon mint céltektonon van legalább 3 db spóra (és nincs, mert nem is lehet rajta gombatest).
  + az objektumok állapotában nem következik be változás
* **Bemenet**

CREATE\_TECTON SemiFertileTecton sft1

SET\_BREAKTIMER sft1 5

PUT\_SPORE SpeedSpore speeds1 sft1

PUT\_SPORE SpeedSpore speeds2 sft1

PUT\_SPORE SpeedSpore speeds3 sft1

ADD\_PLAYER Mycologist mycologist1

CREATE\_MYCELIUM Mycelium m1

ADD\_MYCELIUM\_TO\_TECTON m1 sft1

START\_GAME

GROW\_MUSHROOMBODY mb1 sft1

ENDTURN

STATE sft1

STATE m1

* **Elvárt kimenet**

ft1: SemiFertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

}

myceliumCapacity int = 1

spores Queue<Spore> = {

speeds1

speeds2

speeds3

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m1

}

occupants List<Insect> = {

}

m1: Mycelium

growing boolean = false

location Tecton = sft1

growTimer int = 0

### TSZ/5. számú teszteset – Gombatest sikeres spórakilövése a gombatest elhelyezkedése szerinti tektonnal szomszédos FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton)

* **Leírás**

Gombatest sikeresen kilövi a spóráit a gombatest elhelyezkedése szerinti tektonnal szomszédos FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton).

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikeres spórakilövése a következő feltételek fennállása mellett: gombatest az utolsó, azaz a harmadik kilövése előtt csak valamely szomszédjára lőhet spórát
  + a gombatest spóráinak száma a kilövés következtében 0-ra csökken
  + a kilőtt spórákat a továbbiakban a céltekton tartja nyilván
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

ADD\_NEIGHBOUR f1 f2

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

ADD\_SPORE SpeedSpore speeds1 mb1

ADD\_SPORE SpeedSpore speeds2 mb1

ADD\_SPORE SpeedSpore speeds3 mb1

START\_GAME

EJECT\_SPORES mb1 ft2

ENDTURN

STATE ft1

STATE ft2

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

speeds1

speeds2

speeds3

mb1-speeds1

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

mb1: MushroomBody

remainingEjects int = 2

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds2

}

### TSZ/6. számú teszteset – Gombatest sikeres spórakilövése olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amely a gombatest elhelyezkedése szerinti tektonnal szomszédos tekton szomszédja

* **Leírás**

Gombatest sikeres, összesen a harmadik (utolsó) spórakilövése olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amely a gombatest elhelyezkedése szerinti tektonnal szomszédos tekton szomszédja.

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikeres spórakilövése a következő feltételek fennállása mellett: gombatest az utolsó, azaz a harmadik kilövése előtt csak valamely szomszédjára lőhet spórát. Fejlettnek az utolsó kilövése során minősül, amely abban nyilvánul meg, hogy ekkor a szomszédja szomszédjára is tud lőni
  + a gombatest a harmadik kilövését követően inaktívvá válik
  + a kilőtt spórákat a továbbiakban a céltekton tartja nyilván
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

CREATE\_TECTON FertileTecton ft3

SET\_BREAKTIMER ft3 5

ADD\_NEIGHBOUR ft1 ft2

ADD\_NEIGHBOUR ft2 ft3

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

ADD\_SPORE SpeedSpore speeds1 mb1

ADD\_SPORE SpeedSpore speeds2 mb1

ADD\_SPORE SpeedSpore speeds3 mb1

SET\_REMAININGEJECTS mb1 1

START\_GAME

EJECT\_SPORES mb1 ft3

ENDTURN

STATE ft1

STATE ft2

STATE ft3

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft3: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

speeds1

speeds2

speeds3

mb1-speeds1

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

mb1: MushroomBody

remainingEjects int = 0

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds2

}

### TSZ/7. számú teszteset – Gombatest spórahiány miatti sikertelen spórakilövése a gombatest elhelyezkedése szerinti tektonnal szomszédos FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton)

* **Leírás**

Gombatest egy-egy spórakilövést kísérel meg az elhelyezkedése szerinti tektonnal szomszédos két FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton). Sorrendben a második kísérlet sikertelen, tekintettel arra, hogy a gombatestnek ekkor már nincsen kilőhető spórája.

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikertelen spórakilövése a következő feltételek fennállása mellett: gombatest az utolsó, azaz a harmadik kilövése előtt csak valamely szomszédjára lőheti ki meglévő spóráit
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

CREATE\_TECTON FertileTecton ft3

SET\_BREAKTIMER ft3 5

ADD\_NEIGHBOUR ft1 ft2

ADD\_NEIGHBOUR ft1 ft3

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

START\_GAME

EJECT\_SPORES mb1 ft2

EJECT\_SPORES mb1 ft3

ENDTURN

STATE ft1

STATE ft2

STATE ft3

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

}

myceliumCapacity int = 1

spores Queue<Spore> = {

mb1-speeds1

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft3: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

mb1: MushroomBody

remainingEjects int = 2

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds2

}

### TSZ/8. számú teszteset – Gombatest sikertelen spórakilövése olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amely a gombatest elhelyezkedése szerinti tektonnal szomszédos tekton szomszédja

* **Leírás**

Gombatest sikertelenül kísérel meg spórakilövést olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amely a gombatest elhelyezkedése szerinti tektonnal szomszédos tekton szomszédja, mert nem ez a gombatest harmadik (összességében az utolsó) spórakilövése.

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest sikertelen spórakilövése a következő feltételek fennállása mellett: gombatest az utolsó, azaz a harmadik kilövése előtt csak valamely szomszédjára lőhet spórát. Fejlettnek az utolsó kilövése során minősül, amely abban nyilvánul meg, hogy ekkor a szomszédja szomszédjára is tud lőni
  + az objektumok állapotában nem következik be változás
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

CREATE\_TECTON FertileTecton ft3

SET\_BREAKTIMER ft3 5

ADD\_NEIGHBOUR ft1 ft2

ADD\_NEIGHBOUR ft2 ft3

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

ADD\_SPORE SpeedSpore speeds1 mb1

ADD\_SPORE SpeedSpore speeds2 mb1

ADD\_SPORE SpeedSpore speeds3 mb1

START\_GAME

EJECT\_SPORES mb1 ft3

ENDTURN

STATE ft1

STATE ft2

STATE ft3

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft3: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

mb1: MushroomBody

remainingEjects int = 3

location Tecton = ft1

mushroomSpores List<Spore> = {

speeds1

speeds2

speeds3

mb1-speeds1

mb1-speeds2

}

### TSZ/9. számú teszteset – Gombatest sikertelen spórakilövése olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton), amely a gombatest elhelyezkedése szerinti tekton harmadik szomszédja

* **Leírás**

Gombatest sikertelenül kísérel meg spórakilövést olyan FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton, céltekton), amely a gombatest elhelyezkedése szerinti tekton harmadik szomszédja. [Azaz létezik A, B, C és D FertileTecton, amelyek a következőképpen szomszédosak (a szomszédosságot a – jelöli): A – B – C – D. (A tektonok egyéb módon nem szomszédosak egymással.) A gombatest A FertileTectonon található. A tekton harmadik szomszédja D tekton.]

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + gombatest az utolsó, azaz a harmadik kilövése előtt csak valamely szomszédjára lőhet spórát. Fejlettnek az utolsó kilövése során minősül, amely abban nyilvánul meg, hogy ekkor a szomszédja szomszédjára is tud lőni. Érettégétől függetlenül azonban nem képes spórakilövésre az elhelyezkedése szerinti tekton harmadik szomszédjára
  + az érettnek minősülő gombatestnek az elhelyezkedése szerinti tekton harmadik szomszédja tekintetében megkísérelt spórakilövését teszteljük
  + az objektumok állapotában nem következik be változás
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

CREATE\_TECTON FertileTecton ft3

SET\_BREAKTIMER ft3 5

CREATE\_TECTON FertileTecton ft4

SET\_BREAKTIMER ft4 5

ADD\_NEIGHBOUR ft1 ft2

ADD\_NEIGHBOUR ft2 ft3

ADD\_NEIGHBOUR ft3 ft4

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

ADD\_SPORE SpeedSpore speeds1 mb1

ADD\_SPORE SpeedSpore speeds2 mb1

ADD\_SPORE SpeedSpore speeds3 mb1

SET\_REMAININGEJECTS mb1 1

START\_GAME

EJECT\_SPORES mb1 ft4

ENDTURN

STATE ft1

STATE ft2

STATE ft3

STATE ft4

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft3: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

ft4

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft4: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

mb1: MushroomBody

remainingEjects int = 0

location Tecton = ft1

mushroomSpores List<Spore> = {

speeds1

speeds2

speeds3

mb1-speeds1

mb1-speeds2

}

### TSZ/10. számú teszteset – Inaktív gombatest sikertelen spórakilövése a gombatest elhelyezkedése szerinti tektonnal szomszédos FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton)

* **Leírás**

Inaktív gombatest sikertelenül próbál spórakilövést végrehajtani a gombatest elhelyezkedése szerinti tektonnal szomszédos FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton; céltekton).

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + Inaktív gombatest semmilyen cselekvésre, így spórakilövésre sem képes (ebbe az állapotba a harmadik spórakilövése után kerül a gombatest, és ekkor már nincsen spórája)
  + az objektumok állapotában nem következik be változás
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

ADD\_NEIGHBOUR ft1 ft2

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

DEACTIVATE mb1

START\_GAME

EJECT\_SPORES mb1 ft2

ENDTURN

STATE ft1

STATE ft2

STATE mb1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 4

neighbours List<Tecton> = {

ft1

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

mb1: MushroomBody

remainingEjects int = 0

location Tecton = ft1

mushroomSpores List<Spore> = {

}

### TSZ/11. számú teszteset – StunSpore sikeres elhelyezése FertileTectonon (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton)

* **Leírás**

Egy StunSpore sikeresen elhelyezésre kerül egy FertileTectonra (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton; céltekton).

(Megjegyzés: SustainingTecton, MultiLayeredTecton, AridTecton és SemiFertileTecton mint céltektonok, valamint SplitSpore, PreventCutSpore, SpeedSpore és SlownessSpore esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + a spóra elhelyezése következtében a céltekton az elhelyezett spórát nyilvántartásba veszi
  + az objektumok állapota egyebekben nem változik
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

PUT\_SPOREStunSpore stuns1 ft1

STATE ft1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 5

neighbours List<Tecton> = {

}

myceliumCapacity int = 1

spores Queue<Spore> = {

stuns1

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

### TSZ/12. számú teszteset – Gombatest inaktívvá válása a harmadik spórakilövését követően

* **Leírás**

Gombatest három spórakilövését követően inaktívvá válik.

Egy FertileTectonon (nem SustainingTecton, nem MultiLayeredTecton és nem AridTecton; céltekton) található gombatest mindhárom alkalommal egy darab SpeedSpore típusú spórát lő ki (funkcionálisan mindig az összes spóráját kilövi; körönként egy-egy SpeedSpore-t termel) különböző FertileTectonokra (fenti jellegű; céltekton). A kilövések során a céltekton mindig szomszédos. A rovar tektonról-tektonra mozogva bejárja a pályát és közben nem eszik spórát, nem rág el fonalat.

(Megjegyzés: SustainingTecton, MultiLayeredTecton és AridTecton mint a gombatest elhelyezkedése szerinti tekton és céltekton, CarnivorousMycelium, valamint SplitSpore, StunSpore, PreventCutSpore és SlownessSpore esetén a teszt hasonlóképpen működik, mutatis mutandis.)

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + a gombatest a harmadik spórakilövését követően inaktívvá válik
  + a többi objektum állapotában bekövetkezett változások vizsgálata
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

CREATE\_TECTON FertileTecton ft3

SET\_BREAKTIMER ft3 5

CREATE\_TECTON FertileTecton ft4

SET\_BREAKTIMER ft4 5

ADD\_NEIGHBOUR ft1 ft2

ADD\_NEIGHBOUR ft1 ft3

ADD\_NEIGHBOUR ft1 ft4

ADD\_NEIGHBOUR ft2 ft3

ADD\_NEIGHBOUR ft3 ft4

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

CREATE\_MYCELIUM Mycelium m1

ADD\_MYCELIUM\_TO\_TECTON m1 ft1

CREATE\_MYCELIUM Mycelium m2

ADD\_MYCELIUM\_TO\_TECTON m2 ft2

CREATE\_MYCELIUM Mycelium m3

ADD\_MYCELIUM\_TO\_TECTON m3 ft3

CREATE\_MYCELIUM Mycelium m4

ADD\_MYCELIUM\_TO\_TECTON m4 ft4

ADD\_PLAYER Entomologist entomologist1

CREATE\_INSECT i1 f1

START\_GAME

EJECT\_SPORES mb1 ft2

ENDTURN

MOVE i1 ft2

ENDTURN

EJECT\_SPORES mb1 ft3

ENDTURN

MOVE i1 ft3

ENDTURN

EJECT\_SPORES mb1 ft4

ENDTURN

MOVE i1 ft4

ENDTURN

STATE ft1

STATE ft2

STATE ft3

STATE ft4

STATE mb1

STATE m1

STATE m2

STATE m3

STATE m4

STATE i1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 2

neighbours List<Tecton> = {

ft2

ft3

ft4

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

m1

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 2

neighbours List<Tecton> = {

ft1

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

mb1-speeds1

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m2

}

occupants List<Insect> = {

}

ft3: FertileTecon

breakTimer int = 2

neighbours List<Tecton> = {

ft1

ft2

ft4

}

myceliumCapacity int = 1

spores Queue<Spore> = {

mb1-speeds2

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m3

}

occupants List<Insect> = {

}

ft4: FertileTecon

breakTimer int = 2

neighbours List<Tecton> = {

ft1

ft3

}

myceliumCapacity int = 1

spores Queue<Spore> = {

mb1-speeds3

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m4

}

occupants List<Insect> = {

i1

}

mb1: MushroomBody

remainingEjects int = 0

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds3

}

m1: Mycelium

growing boolean = false

location Tecton = ft1

growTimer int = 0

m2: Mycelium

growing boolean = false

location Tecton = ft2

growTimer int = 0

m3: Mycelium

growing boolean = false

location Tecton = ft3

growTimer int = 0

m4: Mycelium

growing boolean = false

location Tecton = ft4

growTimer int = 0

i1: Insect

location = ft4

maxMoves = 2

remainingMoves = 2

sporesEaten = 0

effectTimer = 0

state = Normal

### TSZ/13. számú teszteset – Rovar által elvágott gombafonál elsorvadása és az elfogyasztott spóra rovarra gyakorolt hatása

* **Leírás**

A rovar által elvágott CarnivorousMycelium 3 kör elteltével elpusztul. Az elfogyasztott StunSpore hatására a rovar 1 körön keresztül semmilyen aktivitást nem képes kifejteni.

Az első körben a rovar ft3-an állva elvágja az ott lévő húsevő gombafonalat (CarnivorousMycelium). A gombatest a harmadik körben kilövi az időközben termelődött spóráit ft2-re. A rovar ugyanebben a körben elfogyasztja az ft6-on található StunSpore-t. Ennek hatása az ötödik kör elejére megszűnik, akkor, amikor az elvágott gombafonál elsorvad és vele együtt a gombatesttel való összeköttetés nélkül maradt gombafonalak is ft4-en és ft5-ön.

* **Ellenőrzött funkcionalitás, várható hibahelyek**
  + a rovar által elvágott gombafonál elsorvadása
  + a rovar által elfogyasztott spóra hatása
  + a többi objektum állapotában bekövetkezett változások vizsgálata
* **Bemenet**

CREATE\_TECTON FertileTecton ft1

SET\_BREAKTIMER ft1 5

CREATE\_TECTON FertileTecton ft2

SET\_BREAKTIMER ft2 5

CREATE\_TECTON FertileTecton ft3

SET\_BREAKTIMER ft3 5

CREATE\_TECTON FertileTecton ft4

SET\_BREAKTIMER ft4 5

CREATE\_TECTON FertileTecton ft5

SET\_BREAKTIMER ft5 5

CREATE\_TECTON FertileTecton ft6

SET\_BREAKTIMER ft6 5

PUT\_SPORE StunSpore stuns1 ft6

ADD\_NEIGHBOUR ft1 ft2

ADD\_NEIGHBOUR ft1 ft3

ADD\_NEIGHBOUR ft1 ft6

ADD\_NEIGHBOUR ft2 ft3

ADD\_NEIGHBOUR ft2 ft6

ADD\_NEIGHBOUR ft3 ft4

ADD\_NEIGHBOUR ft4 ft5

ADD\_PLAYER Mycologist mycologist1

CREATE\_MUSHROOMBODY mb1 ft1

CREATE\_MYCELIUM Mycelium m1

ADD\_MYCELIUM\_TO\_TECTON m1 ft1

CREATE\_MYCELIUM Mycelium m2

ADD\_MYCELIUM\_TO\_TECTON m2 ft2

CREATE\_MYCELIUM Mycelium m3

ADD\_MYCELIUM\_TO\_TECTON m3 ft4

CREATE\_MYCELIUM Mycelium m4

ADD\_MYCELIUM\_TO\_TECTON m4 ft5

CREATE\_MYCELIUM Mycelium m5

ADD\_MYCELIUM\_TO\_TECTON m5 ft6

CREATE\_MYCELIUM CarnivorousMycelium cm1

ADD\_MYCELIUM\_TO\_TECTON cm1 ft3

ADD\_PLAYER Entomologist entomologist1

CREATE\_INSECT i1 ft5

START\_GAME

ENDTURN

MOVE i1 ft4

MOVE i1 ft3

CUT i1

ENDTURN

ENDTURN

MOVE i1 ft2

MOVE i1 ft1

ENDTURN

EJECT\_SPORES mb1 ft2

ENDTURN

MOVE i1 ft6

EAT i1

ENDTURN

ENDTURN

ENDTURN

STATE ft1

STATE ft2

STATE ft3

STATE ft4

STATE ft5

STATE ft6

STATE mb1

STATE m1

STATE m2

STATE m5

STATE i1

* **Elvárt kimenet**

ft1: FertileTecon

breakTimer int = 1

neighbours List<Tecton> = {

ft2

ft3

ft6

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = mb1

mycelia Queue<Mycelium> = {

m1

}

occupants List<Insect> = {

}

ft2: FertileTecon

breakTimer int = 1

neighbours List<Tecton> = {

ft1

ft3

ft6

}

myceliumCapacity int = 1

spores Queue<Spore> = {

mb1-speeds1

mb1-speeds2

mb1-speeds3

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m2

}

occupants List<Insect> = {

}

ft3: FertileTecon

breakTimer int = 1

neighbours List<Tecton> = {

ft1

ft2

ft4

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft4: FertileTecon

breakTimer int = 1

neighbours List<Tecton> = {

ft3

ft5

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft5: FertileTecon

breakTimer int = 1

neighbours List<Tecton> = {

ft4

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

}

occupants List<Insect> = {

}

ft6: FertileTecon

breakTimer int = 1

neighbours List<Tecton> = {

ft1

ft2

}

myceliumCapacity int = 1

spores Queue<Spore> = {

}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {

m5

}

occupants List<Insect> = {

i1

}

mb1: MushroomBody

remainingEjects int = 1

location Tecton = ft1

mushroomSpores List<Spore> = {

mb1-speeds4

mb1-speeds5

}

m1: Mycelium

growing boolean = false

location Tecton = ft1

growTimer int = 0

m2: Mycelium

growing boolean = false

location Tecton = ft2

growTimer int = 0

m5: Mycelium

growing boolean = false

location Tecton = ft6

growTimer int = 0

i1: Insect

location = ft6

maxMoves = 2

remainingMoves = 2

sporesEaten = 1

effectTimer = 0

state = Normal