Grupa 9

ANALIZA CZASOWNIKOWO-RZECZOWNIKOWA

Tworzymy wirtualny świat na planszy typu Hex-a. W świecie będą organizmy posiadające swoje właściwości:

•Siła -mówiąca o zdolności bitewnej naszego organizmu.

•Inicjatywa -mówiąca o kolejności wykonywania ruchu podczas tury.

•Wiek -mówiący ile tur przeżył organizm.

Ponad to organizmy dzielimy na Rośliny i Zwierzęta. Rośliny nie mogą się poruszać ale mogą się rozmnażać samoczynnie.

Zwierzęta zaś mogą się poruszać a przy spotkaniu z innym zwierzęciem na jednym polu, dochodzi do interakcji:

•Zwierzę & zwierzę (różnych gatunków) -Walczy i wygrywa to o większej sile.

•Zwierzę & zwierzę (tego samego gatunku) - rozmnażają się i powstaje zwierzę tego gatunku.

•Zwierzę & Roślina -Jeśli zwierzę jest roślinożercą to zjada Roślinę.

Ponad to jest organizm 'człowiek' którym steruje aktor. Nasz człowiek posiada specjalne umiejętności takie jak:

•Z pokonanych przeciwników podnosi przedmiot.

•Może stworzyć z przedmiotów już posiadanych inne przedmioty kosztem jednej tury wytworzenia ich.

Przedmioty te służą do poprawy statystyk człowieka, oraz przyznania bonusów takich jak:

•magiczny napój (Zwiększa siłę o 10 na 5 tur)

•Błogosławieństwo Teutatesa (w przypadku konfrontacji z silniejszym przeciwnikiem nie umiera, a przesuwa się na sąsiednie wolne pole).

Tylko jeden bonus może być aktywny.

Wynikiem symulacji są :

•Wyniki starć

•Stan populacji

LEGENDA:

X – obiekty

Y – czynności

Z – pola

Ψ – warunki

Aktor – aktor

KARTY CRC

|  |  |
| --- | --- |
| World | |
| Extends: none | |
| Heritage: none | |
| Responsibilities | Collaborators |
| -organisms : List<List<organism>> | Organism |
| +getCoords(enum direction) : (int,int) |  |
| -generateWorld() : List<List<organism>> |  |
| -makeOrganism(enum Species, (int,int)) : Organism |  |
| +getField() : (int,int) |  |
| +getPopulation() : List<organism,int> |  |
|  |  |
|  |  |



|  |  |
| --- | --- |
| GameObject | |
| Extends: none | |
| Heritage: none | |
| Responsibilities | Collaborators |
| -name : string | Human |
| -type : enum | Animal |
| -description : string |  |
| +getName() : string |  |
| +getType() : enum type |  |
| +getDescription() : string |  |

|  |  |
| --- | --- |
| *interface Interactions* | |
| Extends: none | |
| Heritage: none | |
| Responsibilities | Collaborators |
| +Interact() : void | Plant |
| +Move() : void | Animal |
| +Multiply() : void | Human |

|  |  |
| --- | --- |
| *Organism* | |
| Extends: none | |
| Heritage: Plant, Animal, Human | |
| Responsibilities | Collaborators |
| -strength : int | World |
| -initiative : int | GameObject |
| -age : int |  |
| -species : enum |  |
| +get(enum value) : int |  |
| +getSpecies() : enum |  |
| -coords : (int,int) |  |
| -dropTable : List<(gameObject,double)> |  |
| +Drop() : gameObject |  |
| +getCoords() : (int,int) |  |

|  |  |
| --- | --- |
| Animal | |
| Extends: Organism | |
| Heritage: none | |
| Responsibilities | Collaborators |
| -type : enum | Interactions |
| +Interact() : void | Plant |
| +Move() : void | Human |
| +Multiply() : void | World |
| +getType() : enum type |  |

|  |  |
| --- | --- |
| Plant | |
| Extends: Organism | |
| Heritage: none | |
| Responsibilities | Collaborators |
| +Multiply() | Interactions |
|  | Animal |
|  | Human |
|  | World |

|  |  |
| --- | --- |
| Human | |
| Extends: Organism | |
| Heritage: none | |
| Responsibilities | Collaborators |
| -Inventory : List <(gameObject,int)> | Interactions |
| +Craft(gameObject) : void | Animal |
| -buff : enum | Human |
| +Take(gameObject) : void | World |
| +getBuff(enum buff) | GameObject |
| +useItem(enum Item) : void |  |