PRORGRAMOWANIE OBJEKTOWE - PROJEKT 1

Michał Łubiński 184603

Projekt jest prostą symulacją wirtualnego świata ze zwierzętami, roślinami i kontrolowanym przez użytkownika człowiekiem.

* zaimplementowany świat gry i jego wizualizacja
* zaimplementowane wszystkie obowiązkowe gatunki roślin i zwierząt wraz z ich rozmnażaniem się
* zaimplementowany człowiek wraz z jego specjalną umiejętnością
* zaimplementowana funkcjonalność zapisu i wczytania stanu gry z pliku

Kilka ogólnych zasad:

- zwierzęta symbolizowane są przez ich pierwszą literę w nazwie (człowiek – 8)

- rośliny symbolizowane są przez wybrane znaki specjalne

- sąsiadami pola P są wszystkie pola stykające się bokiem lub rogiem z polem P

- jeśli przy kopulacji dwóch zwierząt tego samego gatunku nie ma wolnego pola w sąsiedztwie żadnego z rodziców dziecko nie rodzi się wcale

- jeśli przy rozsiewaniu się roślina spróbuje zająć pole na którym jest już roślina tego samego gatunku kolizja między nimi nie zachodzi i akcja danej rośliny kończy się

- współrzędne w grze zazwyczaj definiuje klasa Wsp której polami są dwa publiczne int’y (publiczne dla wygody by nie pisać „getX/getY” za każdym razem, operowanie publicznymi wartościami nic nie zmienia w tym wypadku a jest bardziej przejrzyste i nie generuje zbędnej ilości kodu)

- gra toczy się dalej po śmierci człowieka, następna tura pod dowolnym klawiszem nie robiącym niczego innego np. „p”.

Zasady działania aplikacji:

W pliku main.cpp zawierana jest wyłącznie klasa KontrolerOrganizmow która z kolei zawiera klasę Swiat oraz wszystkie istniejące organizmy. W klasie tej jest metoda odpowiadająca za wczytanie zapisu z pliku do gry.

Klasa Swiat jest głównym węzłem całej aplikacji, zawiera ona klasy Organizm, Plansza i LogWalki (klasa odpowiadająca za napisy pod wizualizacją symulatora). Jest w niej lista odpowiadająca za organizmy aktualnie żyjące, stos dla organizmów zabitych w aktualnej turze oraz dla organizmów w aktualnej turze urodzonych. Poza tym stos „logów” oraz wskaźnik na obiekt planszy. Są tu również wszystkie metody odpowiedzialne za obsługę organizmów oraz za kontakt między organizmami a planszą dla ustalania takich danych jak losowe ruchy na sąsiednie pola czy zajętość takowych. Jest tu również metoda tury, zapisu stanu gry do pliku oraz wywołania rysowania świata.

Plansza jest klasą odpowiedzialną za obsługę położenia organizmów, ich sąsiedztwa, losowania ruchów oraz znajdywania wolnych pól dla rodzenia czy ucieczki. Głównym źródłem danych o położeniu organizmów jest 2-wymiarowa macierz zawierająca wskaźniki do organizmów znajdujących się na liście w klasie Swiat. Są tu metody potrzebne do obsługi usuwania i wstawiania organizmów na planszę jak i metody generujące losowych sąsiadów oraz metoda znajdująca losowe wolne pole sąsiednie. W tym jest ważna metoda rotacjaSasiadow() która „skacze” od sąsiada do sąsiada tak by przy pełnym obiegu zwrócić każde sąsiednie pole 1 raz.

Klasa Organizm poza typowymi polami siły inicjatywy itp. Posiada pola polog, buff, zywy, typ oraz dwie zdefiniowane wartości TYP\_ZWIERZE i TYP\_ROSLINA używane do rozróżniania czy dany organizm jest rośliną czy zwierzęciem. Polog jest polem całkowicie autorskim a użyte zostało by przyhamować rozmnażanie się zwierząt w zbyt dużym tempie. W momencie narodzin dziecko w następnej turze próbowało rozmnażać się z rodzicami co generowało duży i nienaturalny wysyp młodych w danym miejscu. Buff jest polem potrzebnym do wszelkiego rodzaju dodatkowych wzmocnień siły, inicjatywy czy innych właściwości, w tym wypadku pole buff dodaje się do domyślnej siły organizmu a w obecnej wersji gry nadawany jest wyłącznie przez Guaranne. Zywy to pole potrzebne by zabity oganizm w danej turze nie wykonywał akcji gdy przyjdzie na niego kolei w liście organizmów. Metoda AktualizujBuffyDebuffy() odpowiada za aktualizowanie się wszelkich buff’ów debuff’ów które w jakiś sposób zmieniają się w czasie (np. cooldown dla Tarczy Azura).

Kolizja w grze wygląda inaczej dla roślin i zwierząt toteż mają one różne metody odpowiadające za domyślną kolizję. Ważne jest jednak że domyślna kolizja zachodzi wyłącznie dla organizmu atakującego oraz wyłącznie jeśli atakujący organizm nie ma specjalnego sposobu ataku i atakowany organizm nie ma specjalnego sposobu obrony.

Specjalną umiejętnością człowieka jest Tarcza Alzura która nie pozwala zwierzętom ani samemu człowiekowi doprowadzić do walki. Zamiast tego każde zwierzę które spróbuje wejść na pole człowieka lub każde zwierzę na które spróbuje wejść człowiek jest przesuwane na losowe sąsiednie pole, jeśli pole to jest już zajęte dochodzi do nowej kolizji między zwierzęciem a innym organizmem. Dla tarczy jest pole czasTarczy w klasie człowieka które można podzielić na 3 stany. Stan 1 gdy czasTarczy jest równy 0, wtedy tarcza jest gotowa do użycia.stan 2 gdy czasTarczy jest pomiędzy czasem regeneracji tarczy i czasem działania tarczy, wtedy tarcza się regeneruje i nie można jej aktywować ani nie chroni ona człowieka.stan 3 gdy czasTarczy jest większy od czasu regeneracji, wtedy tarcza jest aktywna. Przy każdej turze dzięki metodzie aktualizujBuffyDebuffy() czasTarczy spada o 1 by zasymulować odpowiednio wszystkie trzy stany. Aktywacja tarczy: podwójne kliknięcie „t”.

Zapis i odczyt gry polega na zapisaniu do pliku tekstowego po kolei żyjących organizmów z listy organizmów ich symboli, współrzędnych i w tym wypadku jednej wartości buff. Następnie wartości te są odczytywane przez klasę KontrolerOrganizmow która porównuje znalezione symbole z symbolami organizmów z własnej listy, generowanej przy inicjalizacji, w której znajduje się po jednym organizmie z każdego gatunku. Dalej dodaje potomka danego organizmu lecz dodatkowo dodaje jeszcze odczytany buff. Klawisz zapisu „s” i wczytania „l” (małe S i L).

Rośliny:

$ - Barszcz Sosnowskiego @ - Guarana \* - Mlecz

„ - Trawa % - Wilcza Jagoda

Zwierzęta: pierwsza litera nazwy (np. W – Wilk, Z - Żółw)

Człowiek: „8”