# Aplikacja Wirtualny Ekosystem

Wirtualny Ekosystem to prosty symulator ekosystemu, który tworzą trzy organizmy – Bakterie, Glony i Grzyby.

Środowisko ekosystemu to prostokątna tablica składająca się z komórek, które nazywane są dalej niszami. Każda nisza może być albo pusta, albo zajęta przez żywy organizm, albo zajęta przez organizm martwy. Środowisko takie wstępnie jest zasiedlane w sposób dowolny organizmami żywymi. Rozwój ekosystemu następuje w kolejnych krokach symulacji. Krok to przejście od bieżącego stanu symulacji do stanu kolejnego. Stan ekosystemu to rozmieszczenie organizmów w niszach. W trakcie tego przejścia organizmy wykonuje swoje funkcje życiowe.

Jeżeli organizm nie jest głodny to stara się rozmnożyć. Organizm szuka w swoim sąsiedztwie wolnej niszy i jeżeli takie są to losowo w jednej z nich umieszcza swojego potomka. Rozmnożenie się zmniejsza licznik posiłków o koszt potomka. Koszt potomka jest charakterystyczny dla każdego gatunku. Potomek jest młody, tzn. jego licznik życia jest ustawiony na wartość początkową i głodny, tzn. licznik posiłków jest wyzerowany. Jeżeli organizm jest grzybem lub bakterią i nie posila się, ani się nie rozmnaża, to zmienia celę, jeżeli w jego sąsiedztwie jest wolna cela. Zmiana następuje w sposób losowy.

Aplikacja została wykonana w języku C++, posiada interfejs konsolowy i wyświetla animację rozwoju ekosystemu.

## Klasa *Organism*

Klasa implementuje wspólne cechy 3 rodzajów organizmów, takie jak życie i jego zmniejszanie, koszt tworzenia potomka, licznik zjedzonych posiłków itp.

## Klasa *SimulationSettings*

W pliku *simulationsettings.h* istnieją typy wyliczeniowe dla rodzajów wypełnień nisz, akcji organizmów do wykonania oraz kierunków w które może udać się organizm.

Klasa utworzona jest zgodnie ze wzorcem Singleton – tylko jedna instancja tej klasy przechowuje ustawienia całej symulacji. Globalnie ustawia symbole dla oznaczeń nisz oraz ilość życia organizmów, a także koszty potomków i limit posiłków.

## Klasa *RandomGenerator*

Klasa zawiera metody które generują liczby pseudolosowe w określonych zakresach i określonego typu. Operowanie na klasie jest możliwe jedynie na metodach statycznych.

## Klasa *Neighborhood*

Klasa informuje o tym co znajduje się w bezpośrednim sąsiedztwie (innych niszach) danej niszy. Operuje ona na kwadratowej indeksowanej tablicy 3x3 która określa zawartość nisz dookoła. Zależnie od potrzeby, klasa posługuje się indeksami numerycznymi albo nazwami z typu wyliczeniowego.

## Klasa *Inhabitant* i *InhabitantIntent*

Klasa *InhabitantIntent* przekazuje informacje o decyzji wirtualnych organizmów, będących reakcją na otaczające ich sąsiedztwo. Klasa *Inhabitant* jest natomiast klasą abstrakcyjną, która zawiera niezbędne metody do komunikacji pomiędzy organizmem a niszą.

## Klasy *Grzyb*, *Glon* oraz *Bakteria*

Klasy *Grzyb*, *Glon* oraz *Bakteria* są potomkami klas *Organizm* oraz *Inhabitant*. Nie definiują własnych pól i metod, ale zawierają wirtualne metody, które implementują ich unikalne zachowanie i akcje w wirtualnym środowisku.

## Klasa *Cell*

Klasa *Cell* to pojedyncza komórka środowiska zawierająca pojedynczy obiekt klasy *Mieszkaniec* lub jej klasy potomnej. Klasa posiada wskaźnik do tej klasy nazwany *occupant* – jeśli wskaźnik wynosi *nullptr* oznacza to że Nisza jest pusta. Posiada metody dzięki którym sprawdza stan lokatora i jego zamiar.

## Klasa *Environment*

Klasa implementuje działanie całej planszy po której poruszają się organizmy. Tworzy prostokątne środowisko poprzez dynamiczną tablicę obiektów *Cell*, kontroluje ilość organizmów, sprawdza czy są martwe, określa sąsiedztwo i wywołuje losowe zmienianie miejsca zalokowania, a także przeprowadza kolejne kroki symulacji. Jest w stanie odczytać początkowy stan środowiska z pliku i zapisać przebieg symulacji.