**ЗМІСТ**

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАВДАННЯ

* 1. Загальні відомості з предметної галузі…………………………………
  2. Призначення розробки та галузь застосування………………………..
  3. Підстава для розробки…………………………………………………..
  4. Постановка завдання……………………………………………………….
  5. Вимоги до програми або програмного виробу………………………….
     1. Вимоги до функціональних характеристик………………………………
     2. Вимоги до інформаційної безпеки…………………………………………
     3. Вимоги до складу та параметрів технічних засобів……………………….
     4. Вимоги до інформаційної та програмної сумісності ……………………

**1.1 Загальні відомості з предметної галузі**

О том, что компьютерные игры – отличное средство не только отдыха, но и обучения. Они способны сформировать у игрока принципы мышления для решения поставленных задач, которые затем возможно применить для решения реальных проблем, и все это – в доступной, игровой форме.

(привести пример Факторио, Майнкрафта и т.п.)

Стратегия, как жанр игр, существовал задолго до создания первых компьютерных игр. Яркий пример – шахматы, пошаговая стратегия.

(описание и про историю RTS-ок)

О том, что ИИ в играх присутствует, как правило, не для быстрой и эффективной победы, а для красивого и эффектного поражения. (пример шутеров, где рефлексы человека не смогут сравится с «жульничающим» ботом.) Или же о том, что в игре очень сложно создать достаточно эффективный алгоритм, дабы тот составил достойную конкуренцию сопернику-человеку. Разработчики имеют обширный арсенал инструментов, дабы сделать действия ИИ-противника более похожим на человеческие. (про скриптинг) (привести пример ботов в стратегиях, что пытаются действовать согласно жестко определенному плану (билдордеру(build order) или посылают юнитов в заранее определенные точки на игровом поле). Однако, строго говоря, ИИ в играх часто не является таковым. (расписать почему не искусственный интеллект).

**1.5. Вимоги до програми або програмного виробу**

**1.5.1. Вимоги до функціональних характеристик.**

Процес введення даних та команд до програми проводиться за допомогою клавіатури та миші. Результуючі значення, що розраховані на основі введених користувачем даних, мають бути приведені до діапазону можливих значень: як-то, користувач не може створити юніта за межами ігрового поля.

Цикл обчислення положень та атрибутів об’єктів також має враховувати ці діапазони. Наприклад, юніти не можуть вийти за межі ігрового поля, а більшість атрибутів не можуть дорівнювати від’ємному значенню.

Виведення даних проводиться у графічному вигляді за допомогою монітору комп’ютера.

**1.5.2. Вимоги до інформаційної безпеки.**

Програма у фінальному вигляді скомпільована та не потребує зв’язку з мережею Інтернет, тому додаткових вимог до безпеки не потребує. Супутні файли (DLL-бібліотеки, файли графічних ресурсів тощо) мають зберігати своє положення відносно виконуваного файлу програми задля уникнення можливих несправностей програми.

**1.5.3. Вимоги до складу та параметрів технічних засобів**

Для користувача є важливим мати систему з відеокартою, що сумісна з OpenGL версії 3.3 або вище. Операційною системою має бути Windows 10 версії 64 bit. Мінімально можлива комплектація ЕОМ:

* CPU: сімейства Intel Celeron;
* GPU: 3D адаптер Nvidia, Intel HD або AMD/ATI;
* RAM: 2048 МБ.
* VRAM (відеопам’ять): 64 МБ;
* Вільне місце на накопичувачі: 128 МБ;

**1.5.4. Вимоги до інформаційної та програмної сумісності.**

(как надо делать или как будет делаться?)

Програма має бути реалізована на мові C++ за допомогою IDE Visual Studio 2019. Допоміжні бібліотеки, використані для розробки:

* RayLib v3.5, безкоштовна бібліотека на мові C з вільним вихідним кодом, що спрямована на допомогу у створені відеоігр.
* OpenMP 2.0, відкритий стандарт та бібліотека для створення застосунків з використанням паралельних обчислень.