

## Вариант 26 Защитные башни II (\*\*\*)

Разработать приложение, позволяющее описывать состояние игры типа Tower Defense, в которое должно входить информация о врагах, защитных постройках, замке и ландшафте, которая хранится в соответствующих описателях.

Ландшафт (уровень) представляет собой прямоугольное поле, разделенное на квадраты: замок (один), логова врагов (одно или несколько), дорога, поле, лес, защитные башни; и таблицу расположения врагов.

Описатель врага содержит название, максимальное и текущий запас здоровья, количество золота, скорость передвижения, и таблицу эффектов. Враги перемещаются от логова по дороге к замку. Расчет текущего здоровья врага осуществляется в зависимости от эффектов и нанесенного ему урона. Если здоровье врага становится меньше или равно 0, то он погибает, замку начисляется запас золота врага. Враг может наносить урон замку в количестве оставшегося здоровья при достижении замка, после этого враг погибает, золото замку не начисляется.

Описатель замка содержит следующую информацию: название замка, текущую и максимальную прочность, количество золота.

Описатель логова содержит таблицу описателей врагов и время их выхода из логова.

Предусмотреть три основных типа башен – простая башня, магическая башня, магическая ловушка.

Описатель башни содержит следующую информацию: уровень башни и таблицу характеристик башни соответствующую уровню: стоимость и радиус действия. Башня может быть построена только в поле.

Простая и магическая башни наносят урон врагам в соответствии с установленной стратегией: ближайший к башне, ближайший к замку, наиболее слабый, наиболее сильный, наиболее быстрый и т.п. Дополнительно в таблице характеристик присутствуют: наносимый урон и скорострельность.

Магические ловушки и магические башни наносят врагам эффекты: башни – при попадании снаряда во врага, ловушки всем врагам в радиусе действия, на которых не наложены эффекты от данной ловушки. Дополнительно в таблице характеристик присутствует описатель эффекта.

Описатель эффекта содержит оставшееся время действия эффекта, тип и величину эффекта. Предусмотреть следующие типы эффектов: замедление (скорость передвижения врага уменьшается на процент величины эффекта), отравление (враг теряет количество здоровья в единицу времени равное величине эффекта), ослабление (урон наносимый врагу увеличивается на процент величины эффекта). Все эффекты суммируются.

Обеспечить выполнение следующих операций, при помощи которых можно реализовать конструктор уровней игры (модификация всех полей), так и процесс игры.

- ❖ Для ландшафта (уровня):
  - получение/изменение размеров игрового поля;
  - получение/изменение типа клетки с заданной координатой в процессе конструирования уровня и в процессе игры;
  - проверка корректности уровня (наличие одного замка и минимум одного логова, от любого логова до замка должна быть проложена дорога)
- ❖ Для врага:
  - сделать очередной ход (приблизиться к замку и рассчитать здоровье);
- ❖ Для башни:
  - повысить уровень башни;
  - нанести урон врагам в соответствии со стратегией;
- ❖ Для замка
  - нанести урон замку
- ❖ Для логова
  - выпустить очередного врага
  - выпустить очередного врага, если подошло его время выхода из логова

### Порядок выполнения работы

1. На основе описания задачи определить состав классов.
2. Разработать иерархию классов и схему их взаимодействия.
3. Для каждого класса определить его состояние и необходимые методы.
4. Разработать и отладить все классы. Приложение для отладки реализовать в виде простой диалоговой программы.
5. Разработать и отладить класс, реализующий работу приложения. Предусмотреть в классе возможность сохранения таблицы в файле и ее восстановления.
6. Повторить разработку контейнерного класса (вместе с необходимыми итераторами), используя стандартную библиотеку STL. Выбор шаблона классов согласовать с преподавателем.
7. Повторить разработку контейнерного класса (вместе с необходимыми итераторами), используя собственные шаблоны классов.
8. Реализовать приложение с использованием средств оконного интерфейса.
9. Реализовать приложение, использующее разработанные классы, по дополнительному заданию.