

Вариант 25 Защитные башни. Ауры врагов (***)

В данной игре поджанра Tower Defense игровой процесс происходит на местности покрытой водоёмами, горами и равнинами. Враги, порождаемые расположенными на карте логовами, пытаются добраться до замка игрока. Если врагу удастся дойти до замка, они наносят ему урон, вычисляемый как количество оставшегося здоровья умноженного на коэффициент урона, после чего погибают. Замок восстанавливает свою защиту со временем в зависимости от скорости ремонта. Цель врагов - уничтожить замок, цель игрока - не позволить им это сделать.

Для защиты замка игрок может установить на равнинах следующие виды построек: защитные башни и стены. Стены просто блокирует проход для врагов, но могут быть разрушены. Башни автоматически открывает огонь по врагам, если они попали в её радиус действия, кроме того, они не могут быть разрушены. Однако необходимо предусмотреть невозможность полного закрытия прохода всем врагам к замку с помощью башен. На установку построек тратится золото, которые периодически приносятся в казну замком.

Враги бывают следующих видов: лёгкая пехота – передвигается по суше по кратчайшему пути от логова до замка, должна обходить стены, с очень небольшой вероятностью может увернуться от попадания; тяжелая пехота – передвигается по суше по кратчайшему пути от логова до замка, может наносить урон стенам на расстоянии используя тяжёлое вооружение с неограниченным боезапасом; авиация – передвигается по кратчайшему пути от логова до замка, должна облетать горы, но может пролетать над постройками и водоёмами, а также может уничтожать пролетаемые стены бомбардировкой, но боезапас ограничен; герой (может быть любого из предыдущих трёх видов) – обладает одной или несколькими аурами, повышающими характеристики врагов в радиусе их действия (скорость, регенерацию, максимальное здоровье и т.п.). Враги могут регенерировать здоровье, но если оно окажется меньше или равно 0, то он погибает.

Кроме установки защитных сооружений, игрок может за золото починить стену, повысить уровень замка (и соответственно улучшить его доходность, прочность и скорость ремонта), повысить уровень башни (тем самым увеличив её урон и дальность стрельбы).

Разработать приложение, позволяющее описывать состояние игры типа Tower Defense, в которое должно входить информация о врагах, защитных постройках, замке и ландшафте. Обеспечить загрузку информации о типах врагов, таблицах улучшений и стоимости построек из конфигурационных файлов, загрузку карты из файлов карты, а также возможность загрузки и сохранения текущего состояния приложения на диск. Для хранения в памяти информации о врагах, защитных постройках и замке используются соответствующие описатели.

Описатель врага содержит название, максимальное и текущий запас здоровья, скорость регенерации и скорость передвижения, коэффициент урона замку, урон стенам (если есть соответствующее вооружение), дальность стрельбы (если тяжёлая пехота), боезапас (если авиация), вероятность избежать попадания (если лёгкая пехота).

Описатель стены содержит следующую информацию: название, максимальная прочность, текущая прочность.

Описатель башни содержит следующую информацию: название, уровень башни и таблицу характеристик башни соответствующую уровню – радиус действия, наносимый урон, скорострельность, стоимость.

Описатель замка содержит следующую информацию: название замка, уровень, текущую прочность, количество золота, таблицу характеристик замка соответствующую уровню – доходность, максимальную прочность, скорость ремонта, стоимость.

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

Ландшафт (уровень) представляет собой таблицу расположения врагов и прямоугольное поле, разделенное на клетки: водоем, горы, равнины. Каждая клетка также может указывать на описатель постройки: защитные постройки, замок или логово врагов.

Обеспечить выполнение как минимум следующих операций, при помощи которых можно реализовать процесс игры:

- ❖ Для ландшафта (уровня):
 - получение/изменение размеров игрового поля;
 - получение/изменение типа клетки с заданной координатой в процессе конструирования уровня (ограничение: от логова до замка должен быть путь для легкой пехоты) и в процессе игры (ограничение: + стены и башни могут быть построены только на равнинах, стена уничтожается, если ее прочность равняется 0);
- ❖ Для любого врага:
 - сделать очередной ход (приблизиться к замку в зависимости от типа, скорости, ландшафта и т.п.);
- ❖ Для таблиц:
 - получить количество строк в таблице;
 - получить характеристики строения по уровню;
 - добавить уровень в таблицу;
- ❖ Для стены:
 - нанести урон стене
 - отремонтировать стену (повысить текущую прочность на x%, но не более максимальной);
- ❖ Для башни:
 - повысить уровень башни;
 - нанести урон ближайшему врагу;
- ❖ Для замка
 - повысить уровень 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, изобразить иерархию классов и схему их взаимодействия, а также состояния и необходимые методы с помощью UML.
2. Разработать, реализовать и отладить основные классы (логова, враги, постройки). Отладку методов можно реализовать с помощью Unit тестирования.
3. Разработку и реализовать контейнерные классы вместе с необходимыми итераторами (таблицы). Выбор шаблона классов согласовать с преподавателем.
4. Разработать и отладить класс, реализующий работу приложения (ландшафт). Предусмотреть в классе возможность загрузки информации из конфигурационных файлов, а также сохранения и загрузки текущего состояния.
5. Реализовать консольное приложение.
6. (*) Реализовать приложение с графическим интерфейсом. В этом случае консольное приложение можно не реализовывать.

Вариант 25 Защитные башни. Ауры врагов (***)

Приложение

Здесь приведён только пример приложения. Для создания карты для игровой сессии можно воспользоваться одним из существующих универсальных редакторов карт (например Tiled). Для реализации прикладной задачи можно добавить недостающие методы.

В данном примере программа работает в два этапа:

1. Загрузка из конфигурационных файлов уровней прокачки башен и стен, описания врагов и аур, расписания выхода врагов из логова, а также ландшафт уровня из файла карты.
2. Режим игры, в котором пользователь может установить башни или стены, прокачать какую-либо из башен или замок, отремонтировать стены. При этом враги порождаются логовом и движутся к замку. Башни в случае попадания врага в радиус действия атакуют его. При этом периодически замок увеличивает размер имеющегося золота, в зависимости от своей доходности. В случае если замок разрушен врагами, пользователь проигрывает. Если все запланированные на уровне враги уничтожены - пользователь выигрывает.

Вариант консольного приложения

Режим игры, представляет собой псевдографическую карту, на которой изображены объекты к примеру следующим образом (можно любым другим):

- . - пустая клетка (равнина);
- ^ - горы;
- ~ - вода;
- @ - замок;
- # - стена;
- a, b, c - обычные враги;
- A, B, C - герои;
- 1, 2, 3 - башни соответствующего уровня;
- * - логово.

Влияние аур на врагов можно изобразить, к примеру, изменением цвета печатаемого символа.

Возможен либо пошаговый режим (в этом случае пользователь может в любой момент управлять выбирать объект для улучшения или установки, а для хода всех объектов необходимо нажать клавишу), либо в реальном времени (в этом случае есть режим движения, в котором объекты совершают ходы с определённым периодом времени, и режим паузы, в котором можно улучшать или устанавливать объекты пользователя).

Вариант графического приложения

Практически аналогичен консольному варианту, но вместо псевдографики и консольного диалога используется какой-либо графический фреймворк (oxugine, Cocos2d, Qt/QML, Allegro). Красота спрайтов и анимации не важна (можно взять любые). В данном случае предполагается только вариант в реальном времени, однако функция паузы не обязательна.