1. Вступление
2. Пару слов об игре — инди-игра в жанре экшен с видом сверху. В ходе игры вам придется стрелять в бесчисленных врагов, перекатываться и опрокидывать столы. Действия всей игры проходят в подземельях ну или в нашем случае их можно назвать лабиринтами. Существует много процедурных генераторов, создающих логичные схемы уровней, обеспечивающие правильный темп игры и вознаграждение игроков, и есть другие генераторы, создающие уровни с петлями и компактные схемы. Но редко можно встретить в одной игре оба типа, именно поэтому мы рассмотрим сегодня эту игру.
3. На данном слайде вы можете увидеть пример типичного уровня игры. На первый взгляд он кажется довольно простым. Как можно заметить он представляет собой лабиринт с точкой начала и выхода, но вместо отсутствующих стен тут проходы между комнатами.

Игрок начинает со входа, пробегает через сундуки  и лавочника , чтобы собрать лут, а затем, наконец, побеждает босса . Если вы уткнулись в длинный тупик, то для быстрого возврата в комнаты можно пользоваться телепортами. Очевидно, что отдельные комнаты созданы вручную, и населены различными врагами, которых нужно расстреливать. При продвижении к следующим уровням игры игрок видит тот же паттерн снова и снова, только этапы становятся больше.

1. Особенность генератора становится заметной, когда вы начинаете играть. Уровни кажутся чуть более спланированными, чем можно ожидать от случайности. Комната с боссом всегда находится на расстоянии от начала. Комнаты с врагами всегда разумно перемежаются спокойными комнатами, лавками и перекрёстками. И самое важное — многие сундуки расположены за петлями с односторонней проходимостью.

Красная линия — это односторонний коридор. Если вы хотите попасть в комнату с сундуком, то нужно проделать долгий путь. Большинство сундуков расположено или в конце односторонней петли или достаточно глубоко внутри уровня, что заставляет игрока с боем проходить множество комнат.

1. Секрет такого прохождения заключается в том, что общие схемы создаются вручную. Вот схема, использованная для генерации показанного выше уровня.

Обычные комнаты (Normal room) — это случайно выбираемые комнаты с врагами, комнаты-перекрёстки или большие комнаты с несколькими выходами. Здесь не показаны «соединительные» комнаты, то есть комнаты без врагов, часто с природными опасностями. Остальные комнаты или задаются заранее, или выбираются из особой таблицы комнат.

В Gungeon существует довольно мало таких схем, называемых «потоками». Это не простые схемы, их дизайн создан на основе определённой особенности, которую можно заметить при многократном прохождении. Это может быть гигантская петля, или важная развилка из множества путей, или необходимость добраться до лавки, чтобы пройти уровень. Они настолько заметны, что спидраннеры заметили различия и составили графики, по которым можно максимально быстро найти босса.

1. Если внимательно посмотреть на схему потока и карту, можно заметить, что они совпадают не полностью. Под лавкой есть дополнительная комната. не соответствующая схеме, и странные комнаты-коридоры.   
   Процесс создания уровня начинается со случайного выбора файла потока наподобие этого. Это структура данных «граф», то есть в ней хранятся взаимосвязи комнат, но не их расположение. Каждая комната содержит метаданные о типе комнаты и соединениях, которые она должна иметь. Соединения имеют направление — каждая схема потока начинается с корневого узла, а затем образует дерево дочерних узлов. Затем дополнительные соединения разбивают структуру дерева для создания петель.
2. Файл потока может преобразовываться ограниченным количеством способов. Во-первых, некоторые специфические комнаты заменятся сериями комнат случайной длины. Эта функция используется только в поздних, более крупных уровнях. Кроме того, некоторые части файла потока имеют альтернативные пути, и один из них выбирается случайным образом. Эта функция используется только дважды.  
   Затем «инъектируются» ещё несколько дополнительных узлов. Эта функция довольно гибка и используется для множества разных целей.  
   Каждая «инъекция» содержит данные, определяющие, какой тип объекта должен быть вставлен, где он должен быть вставлен, вероятность создания и любые условия, которые должны быть выполнены.   
   На этом же этапе генератор выбирает конкретную комнату для каждого узла. В основном это зависит от текущего этапа и необходимого типа комнаты. Существует огромный список комнат — почти 300 для первого этапа — но генератор старается не выбирать одинаковую комнату дважды.  
   Узлы типа «соединитель» работают иначе. Их комнаты выбираются позже, пока создаётся схема. Часто это длинные и узкие комнаты, поэтому очень важно выбрать комнату с правильной ориентацией.
3. После завершения создания потока он разбивается на «составные объекты». Каждый составной объект — это или отдельная петля из комнат, или набор связанных комнат без петель. Это реализуется поиском наименьшей петли на карте и вырезанием её как составного объекта. Операция повторяется, пока на карте не останется петель. Остальная часть карты становится набором разделённых деревьев и соединениями между отдельными составными объектами.

Затем каждый составной объект создаётся по отдельности, на отдельной карте. Соединены вместе они будут позже.

Для планирования составного объекта первая комната размещается в произвольном месте. Затем к схеме одна за другой добавляются комнаты выбором пары выходов, один из которых относится к новой комнате, а другой — к имеющейся схеме. Выходы — это заранее заданные локации в метаданных каждой комнаты. Затем новая комната размещается так, чтобы её выход непосредственно соединялся с выходом из предыдущей комнаты. Потом процесс повторяется.

Если конкретнее, то составные объекты деревьев размещаются при помощи обхода дерева поиском в глубину. Алгоритм выбирает только те пары выходов, которые приводят к появлению новой комнаты без пересечения с предыдущими. В общем случае алгоритм предпочитает выбирать выходы, расположенные далеко от уже существующих. Если разместить комнату невозможно, то он вернётся назад и заново сгенерирует выбор комнат, повторяя этот процесс до трёх раз.

1. На этом этапе небольшие отдельные части подземелья соединены друг с другом, но сами составные объекты нужно связать для создания полной карты. Как видите, оставшиеся соединения в данном случае дают не особо большой выбор. Но могут встречаться и более сложные случаи, чем ранее. Алгоритм обходит карту, начиная с комнаты с наибольшим количеством соединений. Как и раньше, для каждого создаваемого соединения выбирается пара выходов. Если две комнаты находятся в разделённых частях карты, то эти две части карты выравниваются для создания короткого пути. В противном случае для создания маршрута используется поиск пути.  
   И на этом создание схемы уровня заканчивается.

Похоже, что основная цель разработчиков заключалась в создании процедурного генератора, обеспечивающего удовлетворительный игровой процесс.   
Особенность генерации в том, что генерируются первыми самые сложные/важные части карты. Генератор делает упор на создание узких петель и коротких коридоров в самых центральных частях уровня, а затем пытается соединить с ними всё остальное.