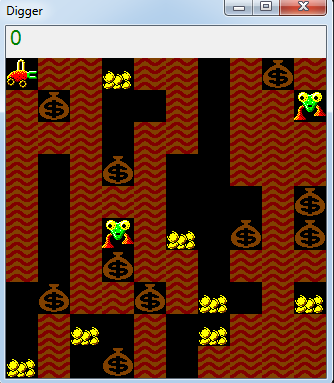


На рисунке видно, что корневым объектом является интерфейс ICreature. На следующем уровне идут Player (игрок), Sack (мешки), Gold (золото), Terrain (земля), Monster(монстры, гоняющиеся за игроком)

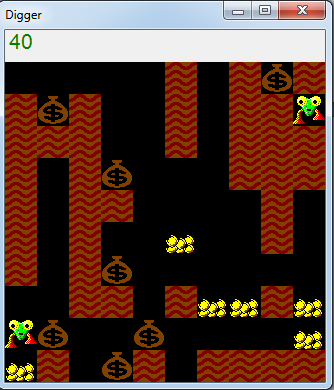
Начало игры



Погоня от монстра



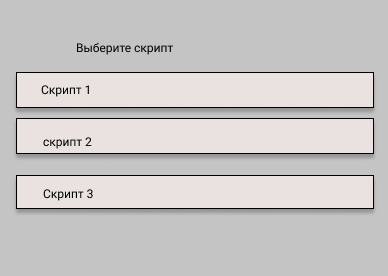
Конец игры, монстр поймал игрока



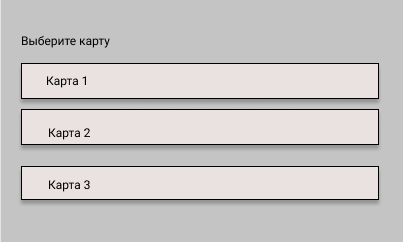
Пример карты без монстров



Форма выбора скрипта



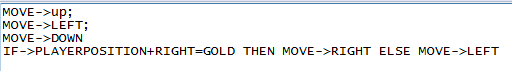
Пример выбора файла карты



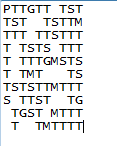
Пример главного экрана



Пример скриптового языка



Пример ввода игровой карты



Функции API

Функции игровых объектов

CreatureCommand Act – на вход подается игровое поле и позиция объекта, на выходе возвращается действие, которое совершит объект.

Bool DeadInConflict – на вход подается объект, который в данный момент по какой-то причине оказывается другой объект, на выходе возвращается логическое значение, true – если объект будет уничтожен и false в противном случае.

GetImageFile – подгружает изображение, которое будет отрисовываться по координатам оюъекта.

GetDrawingPriority – возвращает число-приоритет, на основе которого строится порядок отрисовки объекта.

Функции GameWindow

TimerTick – поддерживает РВ работу программы.

OnKey(Up, Down, Right, Left) – обработка нажатия кнопок управления (на первое время планируется оставить возможность ручного управления игроком).

OnPaint – обрабатывает все объекты, выполняя их команды.

Функции LanguageInterpreter

Command ParseString – на вход подается строка из кода пользователя, возвращается команда для объекта игрока.

LoadMap – подгружает из файла карту, созданную пользователем.

Command[] ParseFIle – создает массив команд из всего кода пользователя