

Курс "Введение в разработку MMO-игр"

Начало курсов: 5 апреля 2008 года, в 14-00, группа до 10 человек.

Целевая аудитория курсов

1. Программисты, которые хотят повысить квалификацию и перейти в область разработки игр
2. Начинающие программисты, которые ищут направление в котором развиваться

Важно: мы не учим конкретному языку программирования. Мы учим образу мышления в программировании. Не важно на каком языке Вы пишете. Важно как. Мы проводим курсы для адаптации новых сотрудников к специфике нашей работы. В том числе и к языкам. В ходе курса будет осуществляться отбор наиболее успешных кандидатов для стажировки в BIT.GAMES — игровом отделении компании BIT Creative.

Формат проведения курсов

Формат проведения курсов — очно-удаленный. Курс будет разбит на несколько (4-7) этапов. Преподаватель встречается со слушателями, дает часть теоретического материала и задание. Срок выполнения задания — неделя. Через неделю — очередная очная встреча, проверка заданий, разбор полетов, следующая часть теоретического материала и практическое задание. Участники курсов получают доступ к закрытому форуму, на котором смогут обмениваться опытом, выкладывать работы, задавать вопросы и получать ответы от преподавателей курсов и сотрудников компании.

Условия поступления на курсы

Перед началом курсов проводится очное собеседование с кандидатами. На курсы зачисляются только те кандидаты, которые успешно прошли собеседование.

Кандидаты на зачисление должны соответствовать следующим требованиям:

1. Знание C++ и ООП
2. Пример своего кода, которым Вы можете гордиться (важен не объем, а качество кода)

Что желательно знать перед прослушиванием курсов

1. Основы компьютерной графики
2. Паттерны проектирования

Литература и ресурсы, с которыми желательно ознакомиться

1. Э.Гамма и др. (GoF) "Приемы объектно-ориентированного проектирования: паттерны проектирования."
2. К.Бек. "Экстремальное программирование - разработка через тестирование". Введение в автоматизированное модульное тестирование.
3. М. Фаулер. "Рефакторинг. Улучшение дизайна существующего кода" (M. Fowler "Refactoring - Improving the Design of Existing Code"). Обязательно к прочтению.
4. М. Фаулер. "Паттерны приложений уровня предприятий" (M. Fowler "Patterns of Enterprise Application Architecture"). Книга позволяет не выдумывать колесо, а сосредоточиться на решении бизнес-задач.
5. Д. Кириевски. «Рефакторинг с использованием шаблонов» (J. Kerievsky. "Refactoring To Patterns"). Рекомендации по рефакторингу систем, а также примеры использования паттернов проектирования при рефакторинге.

План курсов

1. Структура клиент-серверного игрового приложения
 1. Сервер

1. GameLoop
2. Сетевой уровень
3. Система анализа пользовательского ввода
4. Система принятия решений (AI – искусственный интеллект)
5. Синхронизация с постоянным источником данных
2. Клиент
 1. GameLoop
 2. Сетевой уровень
 3. Система анализа пользовательского ввода
 4. Подсистема видео вывода
2. AI – искусственный интеллект
 1. Основные подсистемы AI
 1. Sensory subsystem
 2. Behaviours subsystem
 3. Pathfinding
 2. Типичные способы представления AI логики
 1. Finite State Machine - пример (Pacman)
 2. Behavior Tree - пример (Halo)
 3. GOAL oriented Planning - пример (FEAR)
 3. Использование скриптовых языков в AI: Lua
3. Сетевой уровень
 1. Linux/BSD: demux(select, kpoll, kqueue)
 2. Кластеризация:
 3. Проблемы рассинхронизации клиента и сервера.
4. Persistence (Синхронизация игры с постоянным источником данных)
 1. Уровень абстракции
 2. Атомарность записи
 3. Хранение данных в БД и периодичность сохранения
 4. Кластеризация серверов БД
5. Клиент
 1. GameLoop
 2. Сетевая прослойка
 3. Основные игровые сущности мира
 4. Подсистема видео вывода: 3d – матрицы, кватернионы и проч.
6. Основные принципы разработки
 1. Тестирование
 2. Постоянный рефакторинг кода
 3. Система ночных билдов как для сервера, так и для клиента
 4. Логирование на нескольких уровнях
 5. Максимальная автоматизация и скриптование всех околопроектных задач
 6. Создание специализированного инструментария для отладки как клиента так и сервера