

## Руководство для программистов по устройству в Targem Games

Прежде всего, мы очень любим играть в компьютерные (и консольные!) игры и получаем огромное удовольствие, создавая их сами. Если вы разделяете нашу любовь к играм и программированию - добро пожаловать к нам в коллектив!

Не знаете как начать свой путь в gamedev-разработке? Есть **множество источников информации**.

- Читайте [тематические сайты](#), посвященные разработке игр
- Учитесь на примерах open source кода

— Открытые [исходники](#) игр от Id Software, [разбор исходников](#) Doom 3

— Частично открытые исходники игр от Valve Software, обширная документация по движку [Source](#).

Ну и конечно, лучший способ начать — сделать пусть небольшую, но **свою игру**. Есть множество бесплатных библиотек и продуктов, которые помогут в этом:

- Среда разработки [Microsoft Visual Studio](#)
- [DirectX SDK](#), примеры и документация
- Библиотека и инструменты для работы со звуком [Fmod](#)
- [Bullet Physics Library](#)
- [RakNet](#) — Multiplayer game network engine

Вот некоторые **инструменты**, которые мы постоянно **используем в работе**:

- C++ — основной язык реализации, среда разработки Microsoft Visual Studio
- Скриптовый язык [Lua](#) в качестве внутреннего игрового скрипта
- [Python](#) — для написания вспомогательных утилит
- [Git](#) + [TortoiseGit](#) — система контроля версий исходного кода
- [Subversion](#) + [TortoiseSvn](#) — система контроля версий игровых данных

Мы используем C++ — не "язык" Visual C++, не C++ Builder, не Managed C++, а именно C++. Если вы новичок в C++, настоятельно советуем прочитать перечисленные ниже книги прежде чем пытаться пройти собеседование:

- [Язык программирования C++](#). Базовый курс, Стенли Б. Липпман, Жози Лажойе, Барбара Му.
- [Эффективное использование C++](#). 55 верных советов улучшить структуру и код ваших программ, Скотт Мейерс.
- [Наиболее эффективное использование C++](#). 35 новых рекомендаций по улучшению ваших программ и проектов, Скотт Мейерс.

[Здесь](#) можно посмотреть дополнительную информацию по книгам.

При написании кода мы следуем рекомендациям, которые описаны в нашем внутреннем документе под названием «Coding style». Этот документ описывает стандарты оформления исходного кода — такие как стиль форматирования, способ документации, наименование переменных, типов и файлов ([пример нашего кода](#)).

На собеседовании будет много вопросов об основах программирования и устройстве компьютера, включая, но не ограничиваясь такими как:

- Что такое «O» большое? Что быстрее -  $O(N)$  или  $O(1)$ ?
- Сколько бит в `int`, как в битовом представлении определить знак числа?
- Что такое потоки, методы их синхронизации?
- Простейшие основы линейной алгебры и аналитической геометрии (векторы, матрицы, скалярное/векторное произведение и т.п.)

Всячески приветствуются бонусные знания:

- Администрирование и архитектура компьютерных сетей
- Базы данных (мы любим [MongoDB](#))
- Работа в Linux (под управлением [Ubuntu](#) работают многие наши сервисы)
- Знание основ компьютерной безопасности ([шифрование, обмен ключами](#))

- Знакомство с Web-разработкой (мы работаем с [PHP](#) и [Django](#))
- Участие в open source проектах

Что делать дальше?

Если вы считаете что готовы к собеседованию, то отправляйте резюме и решение [задачи](#) на адрес [job@targem.ru](mailto:job@targem.ru). И не стесняйтесь похвастаться примерами своего кода. Удачи!