**Якубенко София С23-703**

**Системное программирование**

Системное программирование - это раздел программирования, в котором сочетаются исследования новых архитектур, алгоритмов, структур данных и др. и деятельность по проектированию, разработке, тестированию и сопровождению, поддержке системного обеспечения (системного ПО), т.е. для создания новых информационных технологий.

Оптимальная платформа для системного программирования зависит от конкретных требований разработчика. Но UNIX-системы (такие как macOS и Linux) зачастую считаются самыми лучшими. Unix – это семейство переносимых, многопользовательских и многозадачных операционных систем. Рассмотрим преимущества:

1. Мощные инструменты командной строки, то есть системы UNIX предоставляют обширный набор командной строки, что помогает работать системному программисту с файлами, данными более результативно и эффективно.
2. UNIX-системы предоставляют нативные системные интерфейсы.
3. Эти системы имеют огромное количество разработчиков и пользователей, следовательно это обеспечивает доступ к большому набору ресурсов для данного программирования. Например, это исходные коды, библиотеки, документации и тд.
4. Они обладают высокой степенью переносимости, что помогает разработчику. Например, помогает при создать ПО, которое может быть запущено на разных платформах.
5. Системы могут предоставить доступ к исходному коду ядра ос, это помогает разработчику понимать внутреннее устройство данной системы, вносить в нее изменения.

На мой взгляд, наиболее подходящим для системного программирования является С. В качестве аргумента хочу сказать, что С широко используется для системного программирования, потому что он близок к машинному коду, у него богатая поддержка библиотек, обладает переносимостью, может быть легко интегрирован с другими языками. В целом, использование языка программирования C может обеспечить разработчика мощным инструментом для создания системного программного обеспечения с высокой производительностью, эффективностью и контролем над ресурсами компьютера.

**Сетевое программирование + веб-программирование.**

Системное программирование – это процесс разработки программного обеспечения, которое взаимодействует с другими компьютерами или устройствами через сетевые соединения.

Веб-программирование - это процесс создания и разработки веб-приложений, сайтов и других интернет-ресурсов с использованием различных технологий и языков программирования.

Оптимальной платформой для этих видов программирования является Node.js, потому что есть высокая производительность, позволяет масштабировать приложения, имеет обширную систему экопакетов, единый язык программирования, поддерживает WebSocket.

В данной платформе следует использовать JvaScript, так как он является основным для нее. Использование JavaScript как единого языка программирования для клиентской и серверной частей приложения позволяет более эффективно разрабатывать и поддерживать код, а также с легкостью обмениваться данными между клиентом и сервером.

**Работа и программирование систем ии**

TensorFlow подходит для них. Это открытая платформа для глубокого обучения. Она предоставляет широкий набор инструментов и библиотек для разработки различных моделей машинного обучения и нейросетей.

Основным языком программирования для работы сTensorflow является Phyton. Python обеспечивает простой и понятный синтаксис, что делает разработку моделей TensorFlow более удобной и доступной для большинства разработчиков. Кроме того, Python имеет обширную экосистему библиотек и инструментов, которые могут быть использованы в совокупности с TensorFlow для разработки более сложных моделей и решений в области машинного обучения и искусственного интеллекта.

**Работа с визуальными средами**

Для работы с визуальными средами используют различные платформы. Самой оптимальной платформой для этого может быть Unity. Благодаря ей у человека есть широкий набор инструментов для создания интерактивных сцен, работы с графикой, анимацией и тд. Позволяет создавать игры и приложения для различных платформ, включая ПК, консоли, мобильные устройства и виртуальные очки. Unity обладает множеством встроенных инструментов и функций, таких как физическая симуляция, анимация, управление ресурсами и другие. Это позволяет разработчикам создавать разнообразные и интерактивные игровые механики и эффекты.

Unity поддерживает скриптинг на языке C#, который является широко распространенным и удобным языком программирования. Это делает разработку проектов более простой и эффективной. C# тесно интегрирован с Unity и предоставляет доступ к большому количеству функций и классов, которые помогают в разработке игрового контента.