

Санкт-Петербургский государственный университет

Кафедра программной инженерии

Группа 23М07-мм

Морозко Иван Дмитриевич

Отчёт по педагогической практике

Санкт-Петербург
2024

В качестве педагогической практики я занимался обучением студентов разработке на языке C++. В процессе работы над открытой платформой для профилирования данных Desbordante, я указывал на различные проблемы в коде, в основном взаимодействуя с трёмя студентами под руководством ассистента кафедры информационно-аналитических систем Г. А. Чернышева.

Волгушев Иван Романович (Технологии программирования, 3 курс)

С Иваном Романовичем мы обсуждали корректную реализацию семантики перемещения. В качестве примера я объяснял почему стоит использовать `std::move` при добавлении элементов в вектор чтобы избежать лишнего копирования. Я объяснял концепцию категорий значений выражений в языке C++, и то как компилятор выбирает нужный конструктор копирования или перемещения в зависимости от данной категории.

В другом месте я описывал плюсы и минусы использования статических полей в классах. Я подчеркивал, что при параллельном запуске нескольких алгоритмов будет возникать состояние гонки за этот статический ресурс.

Так же с Иваном Романовичем мы обсудили реализацию разделяемого указателя, и то как его передача в функцию по ссылке/по значению влияет на счётчик ссылок.

Наконец, мы обсуждали отказ от приведений типов в стиле языка C в пользу `static_cast`. Я объяснял, что это более безопасный и современный способ привести тип, и что стандарт C++ рекомендует именно его.

Синельников Михаил Алексеевич (Технологии программирования, 3 курс)

В работе с Михаилом Алексеевичем мы разбирали то как ключевые слова `static` и `inline` влияют на правило одного определения. Мы обсуждали внутреннюю линковку и концепцию единицы трансляции.

Так же я указал на возможности нового стандарта C++20 — использование оператора трёхстороннего сравнения `<=>`. Я предложил Михаилу

Алексеевичу вынести всю логику сравнения в один оператор. Данное решение позволило устранить дублирование кода и сделало код более удобным для чтения.

Так же мы обсудили как стоит использовать семантику перемещения в конструкторе при инициализации объектов. Вместе мы рассмотрели три возможных варианта:

- Передача по значению и перемещение
- Передача по универсальной ссылке и прямая передача (*perfect forwarding*)
- Передача по константой ссылке и копирование.

Я описал плюсы и минусы каждого подхода, сославшись на главу из книги Скотта Мейерса «Эффективный и современный C++. 42 рекомендации по использованию C++11 и C++14», в которой автор задаётся таким же вопросом.

Черников Антон Александрович (Математическое обеспечение и адм. инф. систем, 2 курс)

С Антоном Александровичем мы разбирали библиотеку диапазонов в стандартах C++17 и C++20. Вместо традиционного цикла с итераторами я предложил перейти на такие современные алгоритмы как `std::ranges::any_of` и `std::ranges::generate_n`, что сделало код короче и более читаемым, поскольку основную логику цикла можно сразу понять по названию соответствующего метода.

Значительную часть работы я посвятил описанию минусов использования полей в классах с типом неконстантой ссылки. В данном классе реализовывался оператор круглые скобки, что делало объект — экземпляр данного класса функциональным объектом. Вместе с Антоном Александровичем мы разработали идею использования лямбда-функции вместо такого объекта, чтобы замкнуть необходимые переменные в анонимном функторе и передать его туда, где требуется данный объект (конструктор `boost::vf2_subgraph_iso`).