

1 Выводы

В этом семестре я познакомилась с новой объектно-ориентированной парадигмой программирования. Это методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования. Курс представлял собой полноценный проект, который мы создавали на протяжении семестра. Каждая лабораторная работа была связана с предыдущей, что позволяло находить и исправлять недочеты прошлых версий. Эта практика очень полезна, так оно позволяет нам учиться работать “на длинных дистанциях”. Мною было сделано 9 лабораторных работ:

1. В первой работе я познакомилась с базовыми понятиями ООП, такими как наследование, полиморфизм и инкапсуляция. Было спроектированы классы фигур, заданные вариантом, в которых использовались перегруженные операторы, дружественные функции и операции ввода-вывода из стандартных библиотек.
2. Во второй работе я спроектировала динамическую структуру данных – список. Объекты передаются в методы контейнера по значению.
3. В третьей работе я познакомилась с технологией умных указателей. контейнер был переписан с использованием `shared ptr`.
4. В четвертой работе я узнала о шаблонах классов и их проектировании. Был построен шаблон динамической структуры данных.
5. В пятой работе у моего контейнера появились итераторы, что облегчило перемещение по списку.
6. В шестой работе был создан аллокатор памяти, помогающий оптимизировать выделение и освобождение памяти. Свободные блоки хранятся в аллокаторе в контейнере второго уровня – стеке.
7. В седьмой работе был запрограммирован контейнер в контейнере. То есть в контейнере первого уровня хранятся контейнеры второго уровня, внутри которых хранятся фигуры. Фигуры внутри контейнера второго уровня отсортированы по возрастанию площади. Если в ходе выполнения программы контейнер второго уровня полностью освобождается от фигур, то он удаляется из контейнера первого уровня.
8. В восьмой работе я познакомилась с параллельным программированием, что позволило осуществлять быструю сортировку несколько иным образом нежели

в классическом варианте: все рекурсивные вызовы теперь выполняются каждый в своем потоке.

9. В девятой работе я узнала о лямбда-выражениях. Теперь действия над контейнером первого уровня генерируются в виде команд, которые помещаются в контейнер второго уровня. В рамках данной работы это было еще полезно и как тестирование: можно было убедиться, что контейнер первого уровня работает корректно.

В итоге выполнения этого проекта я получила хорошие навыки программирования и проектирования на C++. Освоила ряд возможностей языка, которые уже к данному моменту понадобились мне при выполнении работ в других курсах. Думаю, что изучение этой парадигмы совершенно точно необходимо каждому современному программисту вне зависимости от того, чем он занимается. Поэтому нельзя останавливаться на достигнутом и нужно продолжить изучение ООП и C++ в частности.