

#### Уд (критические темы прошлого семестра)

1. Указатели и ссылки – сходства, отличия, зачем нужны.
2. Сложность алгоритма. Сортировка за  $O(N^2)$ ,  $O(N\log N)$ .
3. Сложность алгоритма. Поиск числа в отсортированном и неотсортированном массиве, доступ к произвольному элементу массива.
4. Модель памяти .flat. Правила работы стека локальных переменных. Правила работы кучи.

#### Хор (основные механики ООП и STL)

1. Инкапсуляция. Что это такое и зачем нужно.
2. Наследование. Что это такое и зачем нужно.
3. Полиморфизм – концепция и примеры.
4. Конструкторы, деструкторы. Когда вызываются, зачем нужны.
5. Виртуальные поля и методы. Пояснить на примере, как работает механика виртуальности.
6. Статические и дружественные поля и методы класса.
7. ООП и память. Правило трех.
8. Схема сборки многофайловой программы – препроцессор, компилятор, линкер.
9. Системы контроля версий – классификация, зачем нужны. Git и Github – основы использования.
10. Стандарты C++. Какие есть, зачем нужны. Краткий обзор.
11. Контейнеры STL – какие есть, на каких структурах данных основаны, для чего применяются.

#### Отл (продвинутые механики ООП и STL)

1. Перегрузка операторов ввода-вывода, инкремента-декремента, присвоения и индексации.
2. Шаблоны (templates) – что это такое, зачем нужны, к каким проблемам приводит их использование.
3. Исключения и умные указатели – что это такое, какие бывают, зачем нужны.
4. Итераторы STL – что это такое и зачем нужны.
5. Абстрактные классы (интерфейсы). Модификаторы методов класса: final, override.
6. const – примеры различного применения (модификатор метода, переменной, указателя, ссылки, итератора).
7. Касты: static\_cast, dynamic\_cast, reinterpret\_cast, const\_cast. Виртуальный деструктор – при чем он тут.

#### 10 (убойные механики ООП и STL)

1. Шаблонизированный аргумент шаблона. Привести пример, аргументировать применение.
2. Лямбда-функции – что это такое, зачем нужны. Лямбды, адаптеры, функторы.
3. Семантика переноса и ссылок rvalue. Правило пяти.
4. Как написать свой итератор и подсунуть в STL. Пример кода и пояснить.
5. Аллокаторы. Что такое, зачем нужны, как написать свой и подсунуть в STL.