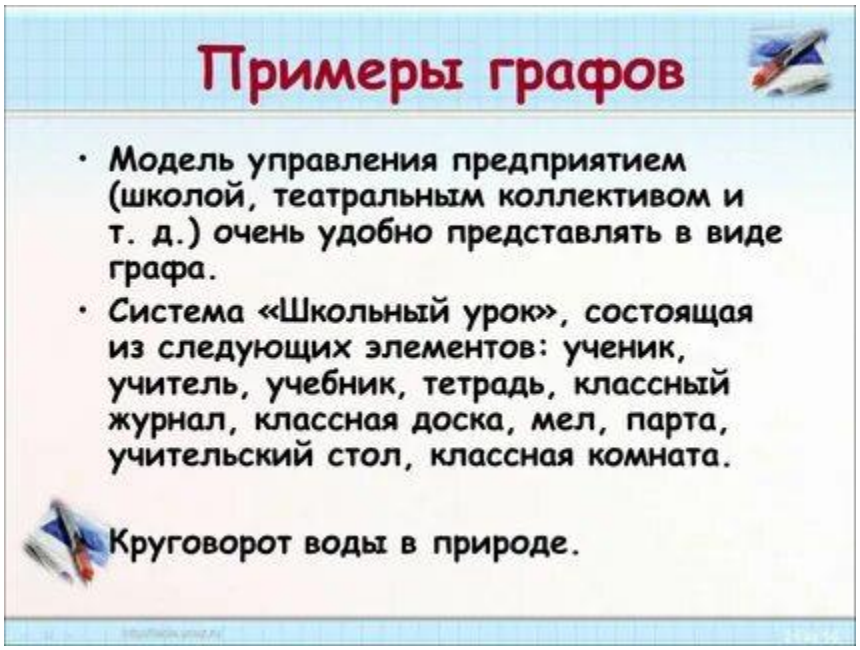
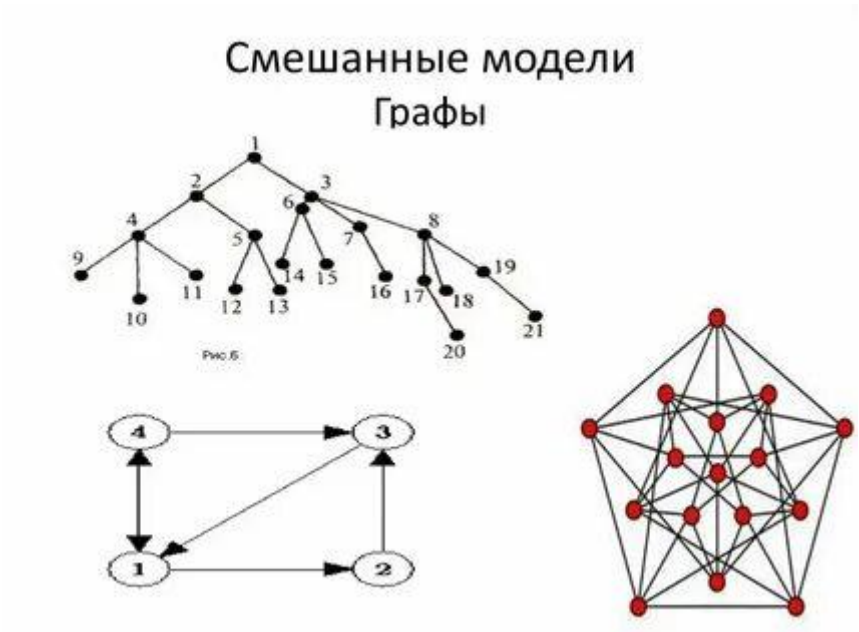
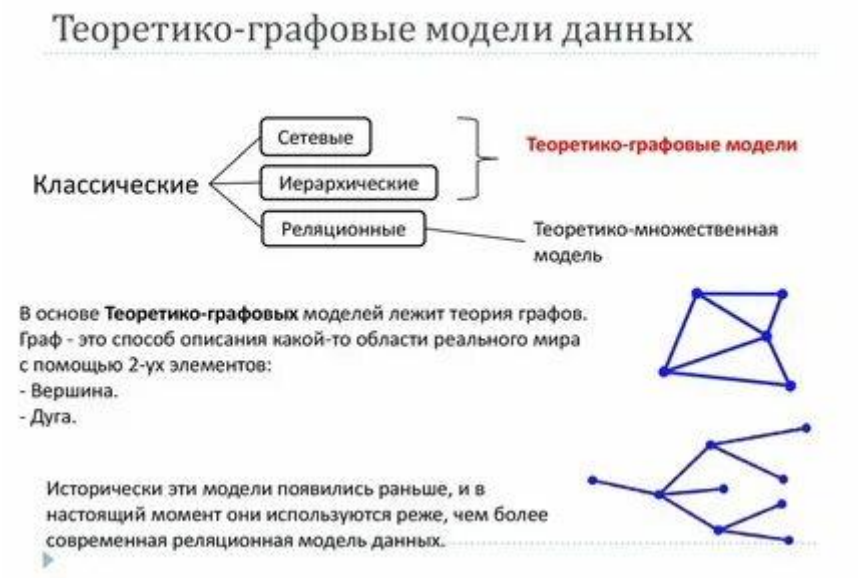
Графовые модели проекта, метрики оценки оттестированности проекта

Графовые модели проекта — это способ представления проекта в виде графа, где задачи или компоненты представлены узлами (вершинами), а зависимости между ними — рёбрами. Такой подход позволяет визуализировать структуру проекта, выявить ключевые зависимости, оптимизировать планирование и мониторинг.







Виды графовых моделей

- **Плоская модель управляющего графа программы (УГП).** Используется для упрощённого представления многокомпонентного объекта. В такой модели компоненты не выделяются отдельно, а их УГП вставляются в общий граф.

- **Иерархическая модель УГП.** Входящие в состав компоненты представлены ссылками на свои УГП, что позволяет учитывать иерархическую структуру проекта.
- **Управляющие графы программы (УГП).** Инструмент для моделирования потока управления в программе. На их основе разрабатываются метрики для оценки оттестированности.

Метрики оценки оттестированности проекта

- **Сложность тестирования V(P,C).** Измеряется максимальным числом неизбыточных тестов, покрывающих все элементы множества компонентов программы M(P,C) по критерию C.
- **Остаточная сложность тестирования DV(P,C,T).** Определяется максимальным числом неизбыточных тестов, покрывающих элементы множества M(P,C), которые остались непокрытыми после прогона набора тестов T.
- **Оценка степени тестированности TV(P,C,T).** Рассчитывается по формуле: $TV(P,C,T) = (V - DV) / V$.
- **Критерий окончания тестирования.** Проект считается достаточно оттестированным, если $TV(P,C,T) \geq L$, где L — заданный уровень оттестированности ($0 \leq L \leq 1$).
- **Покрытие узлов и рёбер графа.** Метрики, которые могут использоваться для оценки полноты тестирования на основе управляющих графов.

Применение графовых моделей в тестировании и оценке оттестированности

- **Анализ требований и определение приоритетов тестирования.** Графы помогают выделить основные функциональные и нефункциональные требования, требующие первоочередного внимания.

- **Планирование тестирования.** Модели позволяют создать детальный план тестирования с учётом зависимостей между компонентами проекта, оптимизировать распределение времени и ресурсов.
- **Автоматизация процесса тестирования.** Создание графа выполнения программы помогает выявить ключевые участки кода, требующие тестирования, и оптимизировать написание и запуск тестов.
- **Мониторинг и контроль.** Графы используются для отслеживания прогресса тестирования, идентификации возможных задержек или проблем.
- **Анализ результатов тестирования.** Графы применяются для выявления областей, требующих дополнительной работы или исправлений.

Таким образом, графовые модели предоставляют инструмент для структурированного анализа проекта и оценки его оттестированности через количественные метрики и визуализация зависимостей.