**52. Размещение программы и данных в памяти**

Программа и данные размещаются в памяти компьютера с использованием сегментов: код, данные, стек и куча. Кодовый сегмент содержит исполняемый код, сегмент данных - статические и глобальные переменные, стек - данные вызовов функций, а куча - динамически выделяемую память.

**53. Виртуальные функции и абстрактные базовые классы**

Виртуальные функции позволяют производным классам переопределять методы базового класса для реализации полиморфизма. Абстрактные базовые классы содержат как минимум одну чисто виртуальную функцию и не могут быть инстанцированы напрямую.

**54. Множественное наследование**

Множественное наследование позволяет классу наследовать поведение и свойства сразу от нескольких классов. Это может привести к проблемам, таким как конфликт имен или "алмазное" наследование, которые решаются с помощью виртуального наследования.

**55. Ассоциативные массивы**

Ассоциативные массивы (словари) хранят пары "ключ-значение" и обеспечивают быстрый доступ к данным по ключу. Используются в языках программирования, таких как Python (dict), C++ (map) и Java (HashMap).

**56. Объекты-функции и предикаты**

Объекты-функции (функторы) – объекты, которые можно вызывать как функции. Предикаты – это функторы, возвращающие логическое значение (true или false) и часто используются в алгоритмах для проверки условий.

**57. Цикл разработки прикладного программного обеспечения**

Цикл включает этапы: концептуализация (определение идеи), анализ (требования), проектирование (структура), кодирование (написание кода), тестирование (проверка), эволюция (доработка) и сопровождение (поддержка).

**58. Критерии оценки качества программы**

Качество программы оценивается по следующим критериям: функциональность, надёжность, производительность, удобство использования, совместимость, переносимость, обслуживаемость и безопасность.

**59. Средства и инструменты разработки программного обеспечения**

К средствам разработки относятся IDE, компиляторы, системы контроля версий, отладчики, тестировочные фреймворки, профилировщики и системы непрерывной интеграции.

**60. Разработка кода программного продукта на основе готовой спецификации на уровне модуля**

Процесс включает: анализ спецификации, проектирование модуля, написание кода, тестирование модуля, документирование и интеграцию с другими модулями.

**61. Ознакомление с технологией тестирования программных продуктов**

Технология включает виды тестирования: модульное, интеграционное, системное, регрессионное и приемочное. Используются автоматические и ручные методы тестирования.

**62. Выполнение отладки и тестирования программы на уровне модуля**

Включает: использование отладчиков для пошагового выполнения, постановку точек останова, анализ переменных, а также написание и запуск модульных тестов для проверки функциональности модуля.

**63. Использование инструментальных средств на этапе отладки программного продукта**

Используются инструменты, такие как отладчики (GDB, LLDB), системы профилирования (Valgrind, gprof), статический анализ кода (Lint, SonarQube) и системы логирования.

**64. Тестирование программного модуля по определенному сценарию**

Сценарий тестирования включает определение тест-кейсов, запуск тестов, сравнение ожидаемых и фактических результатов, а также документирование обнаруженных дефектов и их последующее исправление.

**Другие ответы:**

**52. Размещение программы и данных в памяти:**

Программа размещается в оперативной памяти компьютера во время её выполнения.

Данные также могут быть размещены в оперативной памяти или в памяти, выделенной для хранения переменных и структур данных.

**53. Виртуальные функции и абстрактные базовые классы:**

Виртуальные функции в C++ позволяют переопределить поведение функции в производных классах.

Абстрактные базовые классы содержат хотя бы одну чисто виртуальную функцию и не могут быть созданы напрямую.

**54. Множественное наследование:**

Множественное наследование позволяет классу наследовать свойства и методы от нескольких базовых классов.

**55. Ассоциативные массивы:**

Ассоциативные массивы представляют собой структуру данных, в которой каждому ключу соответствует определенное значение.

**56. Объекты-функции и предикаты:**

Объекты-функции - это объекты, у которых определен оператор вызова функции () , что позволяет их использовать как функции.

Предикаты - это функции, возвращающие логическое значение (истина или ложь) и часто используемые в условных операторах.

**57. Цикл разработки прикладного программного обеспечения:**

Концептуализация

Анализ

Проектирование

Кодирование

Тестирование

Эволюция

Сопровождение

**58. Критерии оценки качества программы:**

Надежность

Производительность

Масштабируемость

Удобство использования

**59. Поддерживаемость**

Средства и инструменты разработки программного обеспечения:

Интегрированные среды разработки (IDE)

Компиляторы

Отладчики

Системы контроля версий

**60. Разработка кода программного продукта на основе готовой спецификации на уровне модуля:**

Разработка функциональности или модуля на основе заданной спецификации или требований.

**61. Ознакомление с технологией тестирования программных продуктов:**

Изучение методов и подходов к тестированию программного обеспечения для обеспечения его качества.

**62. Выполнение отладки и тестирования программы на уровне модуля:**

Исправление ошибок (багов) и проверка корректности работы отдельных модулей программы.

Использование инструментальных средств на этапе отладки программного продукта:

Использование отладчиков, профилировщиков и других инструментов для проверки и улучшения работы программы.

Тестирование программного модуля по определенному сценарию:

Проверка функциональности модуля путем запуска определенных сценариев, входных данных и ожидаемых результатов.