

ВОПРОСЫ К ЭКЗАМЕНУ

- ✓ 1. Объектно-ориентированное программирование как идеология программирования и как технология. Достоинства и недостатки.
- ✓ 2. Основные механизмы объектно-ориентированного программирования.
- ✓ 3. Основные понятия объектно-ориентированного программирования — класс, объект, поле, метод.
- ✓ 4. Класс. Синтаксис объявления класса. Элементы класса. Ссылка (указатель) на созданный объект класса.
- ✓ 5. Конструкторы. Функциональное назначение. Виды конструкторов и их свойства.
- ✓ 6. Конструкторы копирования. Глубокое и поверхностное копирование.
- ✓ 7. Деструкторы. Функциональное назначение. Свойства деструкторов.
- ✓ 8. Объекты и их жизненный цикл. Создание и уничтожение объектов.
- ✓ 9. Инкапсуляция. Спецификаторы доступа. Пример применения.
- ✓ 10. Инкапсуляция. Скрытие элементов класса. Пример.
- ✓ 11. Инкапсуляция. Управление областями видимости элементов класса. Пример.
- ✓ 12. Статические элементы класса. Пример применения.
- ✓ 13. Способы построения новых классов из существующих ранее. Виды наследования.
- ✓ 14. Простое наследование. Доступ при наследовании. Пример.
- ✓ 15. Наследование. Конструктор по умолчанию. Вызов в явном виде конструктора наследуемого класса (инициализация при наследовании).
- ✓ 16. Наследование. Проблема тождественности имен членов классов (доступ к переопределяемым методам). Пример.
- ✓ 17. Наследование. Вызов конструкторов и деструкторов при простом наследовании. Пример применения.
- ✓ 18. Наследование. Правила наследования различных методов.
- ✓ 19. Множественное наследование. Достоинства и недостатки. Пример применения.
- ✓ 20. Множественное наследование. Вызов конструкторов и деструкторов при множественном наследовании.
- ✓ 21. Множественное наследование: проблемы и способы их разрешения.
- ✓ 22. Наследование. Косвенный базовый класс. Ромбовидное наследование.
- ✓ 23. Полиморфизм. Определение. Статический и динамический полиморфизм.
- ✓ 24. Виртуальные методы. Механизм позднего связывания. Пример применения.
- ✓ 25. Виртуальные деструкторы. Пример.
- ✓ 26. Абстрактный класс. Пример.
- ✓ 27. Механизм позднего связывания. Чисто виртуальная функция. Пример.
- ✓ 28. Таблица виртуальных методов. Выбор виртуальной функции при позднем связывании.
- ✓ 29. Дружественность. Определение. Свойства.
- ✓ 30. Дружественные функции. Пример применения.
- ✓ 31. Дружественные классы. Пример применения.
- ✓ 32. Спецификатор friend. Пример.
- ✓ 33. Перегрузка методов. Пример.
- ✓ 34. Перегрузка операций. Унарные операции. Пример.
- ✓ 35. Перегрузка операций. Бинарные операции. Пример.
- ✓ 36. Перегрузка операций. Пример перегрузки операторов ввода/вывода.
- ✓ 37. Шаблоны функций. Синтаксис определения шаблона функции, правила использования. Спецификация шаблона.
- ✓ 38. Шаблоны функций. Пример шаблона функции, использующей несколько типов.
- ✓ 39. Шаблоны классов. Определение. Пример.

- ✓ 40. Достоинства и недостатки шаблонов. Пример.
- ✓ 41. Исключительные ситуации. Способы обработки исключительных ситуаций. Пример.
- ✓ 42. Исключительные ситуации. Перехват исключительных ситуаций. Пример.
- ✓ 43. Исключительные ситуации. Исключения в конструкторах и деструкторах. Пример.
- ✓ 44. Потоки ввода-вывода. Управление потоками.
- ✓ 45. Потоки ввода-вывода. Стандартные потоки ввода/вывода.
- ✓ 46. Потоки ввода-вывода. Неформатный ввод/вывод.
- ✓ 47. Манипуляторы. Предназначение.
- ✓ 48. Ошибки потоков ввода/вывода.
- ✓ 49. Потоки ввода-вывода. Достоинства и недостатки.
- ✓ 50. Файловый ввод-вывод. Пример.
- ✓ 51. Потоки ввода-вывода. Пример использования write().
- ✓ 52. Потоки ввода-вывода. Пример использования read().
- ✓ 53. Потоки ввода-вывода. Пример прямого доступа.
- ✓ 54. Строковые потоки. Пример.
- ✓ 55. Преобразование типов данных. Концепция типа данных. Явное и неявное преобразование типов.
- ✓ 56. Область видимости имен. Выбор области видимости.
- ✓ 57. Класс. Объект. Пример.
- ✓ 58. Иерархия. Понятие «ромбовидное наследование». Пример.
- ✓ 59. Отношения между классами. Пример.
- ✓ 60. Видимость элементов класса. Пример.
- ✓ 61. Библиотека STL. Виды контейнеров.
- ✓ 62. Итераторы. Алгоритмы. Определения и пример.
- ✓ 63. Язык UML. Назначение и основные характеристики языка. Пример модели (привести любой).
- ✓ 64. Язык UML. Нотация языка. Сущности (структурные, поведенческие, прочие) и отношения.
 - 65. Язык UML. Виды и назначение диаграмм UML обзорно – группы диаграмм (структурные, поведенческие, вариантов использования, прочие), их назначение, назвать наиболее распространенных представителей из групп.
 - 66. Язык UML. Диаграмма вариантов использования. Пример.
 - 67. Язык UML. Диаграмма классов. Пример.
 - 68. Язык UML. Поведенческие диаграммы – деятельности или последовательности. Пример.
 - 69. OOD. Понятия связности и зацепления. Сильная связность и слабое зацепление. Закон Деметры.
 - 70. OOD. Замена наследования композицией. Пример.
 - 71. OOD. Принципы SOLID. Принцип единой обязанности.
 - 72. OOD. Принципы SOLID. Принцип открытости/закрытости.
 - 73. OOD. Принципы SOLID. Принцип подстановки Лисков.
 - 74. OOD. Принципы SOLID. Принцип разделения интерфейсов.
 - 75. OOD. Принципы SOLID. Принцип инверсии зависимостей.**
 - 76. OOD. Шаблоны GRASP обзорно – назначение, в какой части жизненного цикла разработки применяются, пример любого из шаблонов.