

- 1) Объекты — это сущности, у которых есть свойства и поведение.
- 2) Объектно-ориентированное программирование (сокращенно ООП) — это парадигма разработки программных систем, в которой приложения состоят из объектов. Примеры систем: наследование, полиморфизм, инкапсуляция.
- 3) Жизненный цикл объекта — это время между его созданием и его уничтожением.
- 4) Объекты в ООП взаимодействуют между собой с помощью сообщений. Принимая сообщение, объект выполняет соответствующее действие. Эти действия обычно называются методами. Примеры: оператор сравнения > оператор сравнения < оператор сравнения ==
- 5) Класс – это способ описания сущности, определяющий состояние и поведение, зависящее от этого состояния, а также правила для взаимодействия с данной сущностью (контракт).

Класс зачастую описывает объект реального мира. Как и реальный объект, класс содержит свой набор параметров и характеристик.

Синтаксис :

Имя класса : Родительские классы{

Private:

Приватные поля и методы

Public:

Публичные поля и методы

Protected:

Защищённые поля и методы

};

6) Чего?

7) Архитектура системы-это концептуальная модель, которая определяет структуру, поведение и другие представления системы. Иерархия: модули классов; базовый класс (класс описывает свойства объектов, содержит множество объектов, подклассов), производный класс (наследует свойства родительского класса и тоже содержит объекты);

8) Чего?

9) Наследование — это возможность создания иерархии классов, когда потомки наследуют все свойства своих предков, могут их изменять и добавлять новые.

```
class имя : [private | protected | public] базовый_класс
{ тело класса };
```

10)

Ключ доступа	Спецификатор в базовом классе	Доступ в производном классе
private	private protected public	нет private private
protected	private protected public	нет protected protected
public	private protected public	нет protected public

- 11) Инкапсуляция – это свойство системы, позволяющее объединить данные и методы, работающие с ними, в классе и скрыть детали реализации от пользователя.
- 12) Полиморфизм в языках программирования и теории типов — способность функции обрабатывать данные разных типов.
- 13) Public : предоставляет доступ к полю или методу внутри и вне класса.  
Private : предоставляет доступ только внутри класса.  
Protected : предоставлять доступ только внутри класса а также в наследуемых классах.
- 14) Конструктор (от construct – создавать) - это особый метод класса, который выполняется автоматически в момент создания объекта класса.  
Деструктор (от destruct - разрушать) - так же особый метод класса, который срабатывает во время уничтожения объектов класса.
- 15) Параметризованный конструктор – это конструктор с параметрами
- 16) Конструктор копирования – это конструктор в параметре которого находится указатель на объект этого же класса.
- 17) Тип\_объекта Имя\_объекта.Поле\_объекта = Данные. Доступ к полям класса соответствует модификаторам доступа этих полей.
- 18) Указатель — это тип данных, хранящий адрес переменной некоторого типа. Ссылка тоже хранит адрес переменной, но её нельзя перенаправить или присвоить нулевой адрес.
- 19) Указатель this – это указатель на объект вызывающий метод
- 20) При присвоении объектов происходит копирование полей из одного объекта в другой.
- 21) Если у метода или функции тип возвращаемых данных не void, то данная функция или метод вернёт объект типа возвращаемого.
- 22) После того как вы определите встроенную функцию с помощью ключевого слова inline, всякий раз когда вы будете вызывать эту

функцию, компилятор будет заменять вызов функции фактическим кодом из функции.

- 23) Дружественной функцией класса называется функция, которая, не являясь его компонентом, имеет доступ к его собственным (private) и защищенным (protected) компонентам. Функция не может стать другом класса "без его согласия".
- 24) Дружественные классы в C++ означают, что все методы одного класса, который объявлен другом, автоматически становятся дружественными методам другого класса, который иницирует дружбу.
- 25) Оператор new выделяет память для хранения значения типа тип\_переменной и возвращает ее адрес. С помощью new могут быть размещены любые типы данных. Оператор delete освобождает память, на которую указывает указатель переменная\_указатель.
- 26) Protected : предоставлять доступ только внутри класса а также в наследуемых классах.
- 27) Множественное наследование-это особенность C++, в которой класс может наследовать от нескольких классов. Конструкторы унаследованных классов вызываются в том же порядке, в котором они наследуются.
- 28) Чтобы использовать базовый класс совместно, просто вставьте ключевое слово virtual в список наследования производного класса. Это создает так называемый виртуальный базовый класс, что означает, что существует только один базовый объект. Этот базовый объект используется всеми объектами в дереве наследования и создается только один раз.
- 29) В объект базового класса можно присвоить объект производного класса.
- 30) Виртуальные функции — специальный вид функций-членов класса. Виртуальная функция отличается об обычной функции тем, что для обычной функции связывание вызова функции с ее определением осуществляется на этапе компиляции. Для виртуальных функций это происходит во время выполнения программы.
- 31) Чисто виртуальная функция – это функция, которая:
  - объявлена в классе со спецификатором virtual
  - объявлена с использованием специального синтаксиса C++, который определяет функцию как не имеющую блока фигурных скобок { }содержащего программный код. Абстрактным классом называется класс, имеющий хотя бы одну чисто виртуальную функцию.
- 32) Перегрузка функций – это создание нескольких функций с одним именем, но с разными параметрами.
- 33) Унарные операции, которые могут быть перегружены - ! & ~ \* + - ++ --
- 34) Во время перегрузки бинарной операции передается ссылка на объект.

- 35) целесообразно перегружать оператор [ ] в классах, где используются массивы.
- 36) Во многих случаях функции имеют аргументы, которые используются настолько редко, что достаточно значения по умолчанию. В таких случаях возможность задания аргументов по умолчанию позволяет указывать только те аргументы функции, которые важны в конкретном вызове.
- 37) Когда адрес перегруженной функции присваивается указателю на функцию, то именно способ объявления этого указателя определяет, адрес какой из перегруженных функций будет присвоен этому указателю.
- 38) Класс вектора стандартной библиотеки C++ является шаблоном класса для контейнеров последовательности. Вектор хранит элементы заданного типа в линейном упорядочении и обеспечивает быстрый произвольный доступ к любому элементу.
- 39) string — класс с методами и переменными для организации работы со строками в языке программирования C++.
- 40) Используется для хранения и извлечения данных из коллекции, в которой каждый элемент является парой, обладающей одновременно значением данных и ключом сортировки. Значение ключа уникально и применяется для автоматической сортировки данных. Multimap : Значение элементов в multimap, но не связанное с ним значение ключа, можно изменить напрямую. Значения ключей, связанные со старыми элементами, необходимо удалить и вставить новые значения ключей, связанные с новыми элементами.
- 41) Static - это ключевое слово в C++, используемое для придания элементу особых характеристик. Для статических элементов выделение памяти происходит только один раз и существуют эти элементы до завершения программы.
- 42) При изменении объявления данных ключевое слово указывает, const что объект или переменная не изменяются.
- 43) Шаблоны классов определяют семейство связанных классов, основанных на типе аргументов, переданных классу при создании его экземпляра. Шаблоны функций похожи на шаблоны классов, но определяют семейство функций. С помощью шаблонов функций можно задавать наборы функций, основанных на одном коде, но действующих в разных типах или классах.
- 44) Шаблоны можно определить в классах или шаблонах классов (в этом случае они называются шаблонами членов). Шаблоны членов, которые являются классами, называются шаблонами вложенных классов.
- 45) Итераторы - это объекты, которые предоставляют последовательный доступ к элементам контейнера. Итератор позволяет перебирать элементы, абстрагируясь от реализации того контейнера, откуда он их берет. Стандартная библиотека предоставляет различные типобезопасные

контейнеры для хранения коллекций связанных объектов. Контейнеры являются шаблонами классов. При объявлении переменной контейнера указывается тип элементов, которые будут храниться в контейнере.

Контейнеры могут создаваться с использованием списков инициализаторов. Они имеют функции элементов для добавления и удаления элементов и выполнения других операций. Контейнеры можно разделить на три категории: последовательные контейнеры, ассоциативные контейнеры и контейнеры-адаптеры.

- 46) Динамическим называется массив, размер которого может изменяться во время исполнения программы. Возможность изменения размера отличает динамический массив от статического, размер которого задаётся на момент компиляции программы.
- 47) В ассоциативных контейнерах элементы вставляются в заранее определенном порядке — , например, как отсортировано по возрастанию. Также доступны неупорядоченные ассоциативные контейнеры. Ассоциативные контейнеры можно объединить в два подмножества: сопоставления (set) и наборы (map).
- 48) Исключение — это событие при выполнении программы, которое приводит к её ненормальному или неправильному поведению. Существует два вида исключений: Аппаратные (структурные, SE-Structured Exception), которые генерируются процессором. К ним относятся, например,
- деление на 0;
  - выход за границы массива;
  - обращение к невыделенной памяти;
  - переполнение разрядной сетки.
- Программные, генерируемые операционной системой и прикладными программами – возникают тогда, когда программа их явно иницирует. Когда встречается аномальная ситуация, та часть программы, которая ее обнаружила, может сгенерировать, или возбудить, исключение.
- 49) Приведение типов в стиле языка C может привести выражение любого типа к любому другому типу данных (исключение это приведение пользовательских типов по значению, если не определены правила их приведения, а также приведение вещественного типа к указателю или наоборот).
- 50) Сигналы - это прерывания, передаваемые процессу операционной системой, которые могут прервать программу преждевременно.
- 51) В с++ используется механизм потокового ввода/вывода. Поток - механизм преобразования значений различного типа в последовательность символов (вывод) и наоборот (ввод), в значение переменной.
- 52) Для форматирования вывода можно использовать функцию printf().