

Лекция №1 по дисциплине  
«Объектно-ориентированное  
программирование» на тему:

**Происхождение и история  
объектно-ориентированного  
программирования. Классы и  
объекты в действительности и в  
программах**

# Цели дисциплины

После изучения дисциплины студенты должны:

- **Знать:**
- Общие принципы объектно-ориентированного программирования;
- Реализацию объектной модели в языке программирования C++;
- Средства обобщенного программирования: классы и шаблоны
- **Уметь:**
- Составлять объектные программы на языке C++;
- Эффективно использовать возможности языка C++ в программах;
- Пользоваться стандартной библиотекой C++
- **Быть ознакомленными:**
- С особенностями объектной модели в других языках программирования;
- С основами объектно-ориентированного проектирования

# Краткое содержание дисциплины

- Классы и объекты. Поля и методы. Конструкторы
- Дружественные функции. Перегрузка операторов
- Производные классы
- Виртуальные функции. Полиморфизм
- Обработка исключений
- Множественное наследование
- Шаблоны функций
- Средства ввода-вывода
- Контейнерные классы и итераторы

## Основная литература

- Х. Дейтел, П. Дейтел. Как программировать на C++. 5-е издание. М.: «Бином», 2018.- 1456 с.
- П. Франка. C++. Учебный курс. 2-е издание. СПб.: «Питер», 2015.- 496 с.
- С. Прата. Язык программирования C++. Лекции и упражнения. 6-е издание. М.: «Диалектика-Вильямс», 2012.- 1248 с.
- С. Липпман и др. Язык программирования C++. Вводный курс. 5-е издание. М.: «Вильямс», 2014.

## Дополнительная литература

- Г.Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. 2-е издание. М.: «Бином», 2014.- 506 с.
- Б.Страуструп. Язык программирования C++. М.: «Бином», 2011.- 1136 с.

# Сложность программного обеспечения

- ❖ Технологии ООП возникли как ответ на нарастающую сложность ПО
- ❖ Не все программные системы сложны.
- ❖ Промышленные программные продукты - диспетчеризация транспорта, складской учет, учет кадров, бухгалтерский учет
- ❖ Промышленная программа имеет такой уровень сложности, что один разработчик не в состоянии охватить все детали системы

# Причины сложности ПО

Сложность ПО вызывается следующими причинами:

## 1. Сложность реального мира

Предприятие – огромный механизм.

Сложность часто возникает из-за нестыковок между пользователями системы и разработчиками

Требования к программной системе меняются в ходе разработки

## 2. Трудность управления процессом разработки

Размер промышленных программных систем – десятки и сотни тысяч строк

- Сопровождение ПО – устранение ошибок в ходе эксплуатации.
- Эволюция ПО – внесение изменений в ответ на изменившиеся требования.
- Сохранение ПО – поддержание работоспособности разрушающейся системы.



## Признаки сложной системы

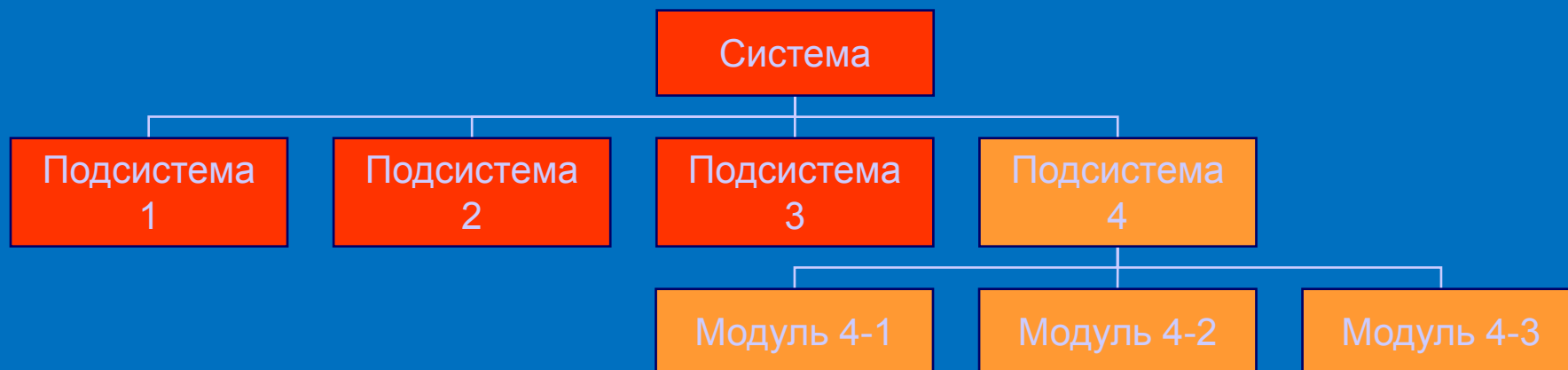
1. Сложные системы являются иерархическими и состоят из подсистем, которые тоже могут быть разделены на подсистемы.
2. Выбор, какие компоненты системы считаются простейшими, зависит от разработчика

## Признаки сложной системы

3. Сложные системы обычно состоят из немногих типов подсистем, но по-разному скомбинированных и организованных.
4. Любая работающая сложная система является развитием работавшей более простой системы

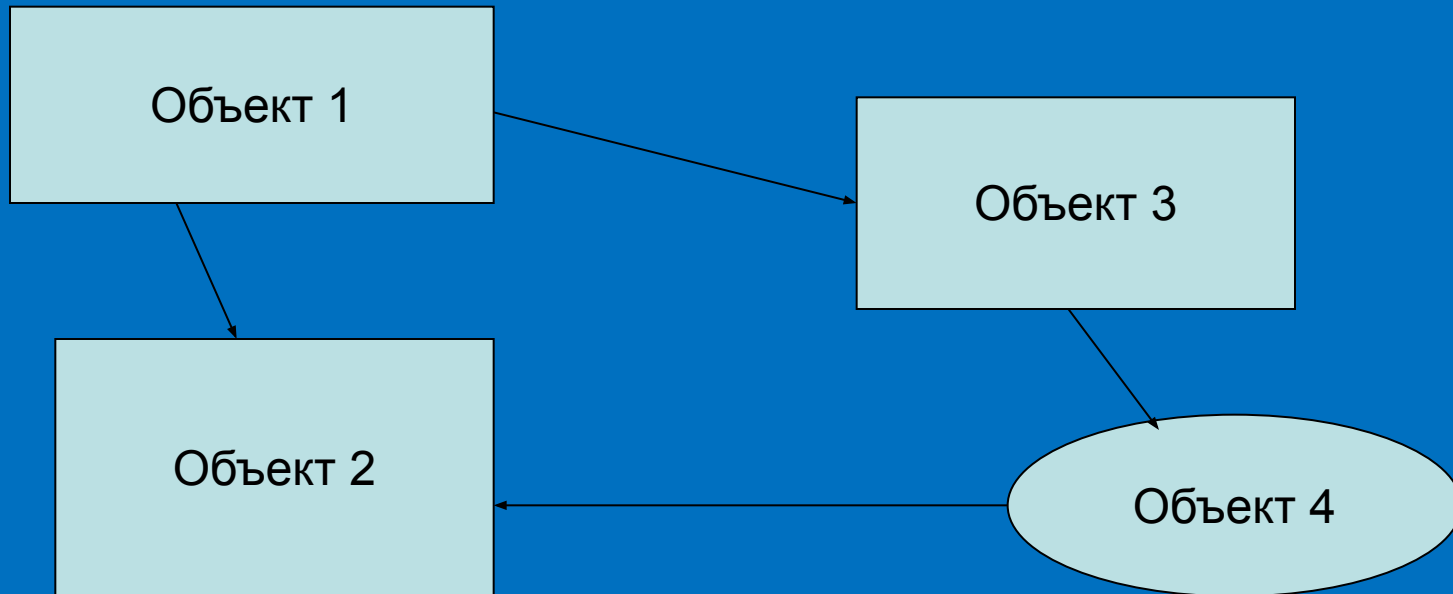
## Алгоритмическая декомпозиция

Это разделение системы на подсистемы и алгоритмы, где каждый модуль выполняет один из этапов



# Объектно-ориентированная декомпозиция

Критерий объектно-ориентированной декомпозиции – принадлежность элементов к различным абстракциям предметной области



# Преимущества объектной декомпозиции

1. Уменьшается размер программных систем за счет повторного использования кода.
  2. Объектно-ориентированные системы более гибки и проще эволюционируют
- **Объектно-ориентированное проектирование** – это методология проектирования, основанная на процессе объектной декомпозиции системы

# Иерархия

Способ, позволяющий снизить уровень сложности – организация **иерархии**

- Необходимо организовать внутри системы иерархию классов и объектов
- Структура классов определяет общность объектов и их поведения внутри системы

# История программирования

- Программирование в машинных кодах
- Процедурные языки программирования  
(FORTRAN, ALGOL)
- Структурное программирование
- Объектно-ориентированное программирование

# Объектно-ориентированное программирование

- Это методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования
- Языки: Simula-67 (впервые введено понятие объекта), Smalltalk-72, 74, 76. Затем объектные концепции перекочевали и в языки высокого уровня, что привело к возникновению Object Pascal, C++, Ада и др. (70-80 годы)