**интерфейс** (совокупность доступных извне абстракции характеристик, состояний и поведения абстракции

**интерфейс-**способ и средства взаимодействия пользователя с программами или программ между собой

**Причины возникновения сложности:**

1. Сложная предметная область

2. Трудность управления

3. Гибкость программы

4. Проблема описания больших дискретных систем

**5 признаков сложных систем:**

1. Сложные системы являются иерархическими и состоят из взаимозависимых систем, которые в

свою очередь могут быть иерархическими и также состоять из систем.

2. Выбор, какие компоненты в данной системе будут элементарными, произволен и в большей

степени зависит от исследователя.

3. Внутрикомпонентная связь элементов выше, чем внешнекомпонентная связь.

4. Иерархические системы обычно состоят из немногих типов подсистем, по-разному

спроектированных и реализованных.

5. Любая сложная система является развитием более простой.

**Принципы объектно-ориентированного программирования**

1. Абстрагирование

2. Ограничение доступа

3. Модульность

4. Иерархическая реализация

(Дополнительные)

5. Параллелизм

6. Типизация

7. Устойчивость

**Абстрагирование** - процесс выделения абстракций. Абстракция это совокупность существенных

характеристик объекта, которые отличают его от других видов, то есть, четко определяют данный

объект с точки зрения дальнейшего рассмотрения и анализа. (Абстракция = класс).

1. Классическая категоризация. Пример: кот может бежать или не бежать. Тогда объект может быть

бегущим или не бегущим.

2. Концептуальная классификация (кластеризация). Кластеризация определяет степень

принадлежности к какому-то классу.

3. ?

**Ограничение доступа** - это сокрытие отдельных элементов абстракции, не затрагивающий ее

существенных характеристик. Необходимость ограничения доступа предполагает выделение двух

частей:

1. Интерфейс (совокупность доступных извне абстракции характеристик, состояний и поведения

абстракции)

2. Совокупность недоступных извне элементов абстракции, включает внутреннюю организацию

абстракции и ее реализацию.

Ограничение доступа позволяет разработчикам:

1. Выполнять конструирование программы по шагам, начиная с разработки интерфейся

(проектировать сверху вниз)

2. Позволяет изменять один элемент абстракции, не затрагивая другие.

**Инкапсуляци**я - это сочетание объединения всех свойств предметов, характеризующих его состояние и поведение в единой абстракции с ограничением доступа к ее реализации.

**Модульность** - принцип разработки программной системы, которая предполагает ее реализацию в отдельных модулях. Модуль - это физический контейнер некоторого набора логических элементов. Следование этому принципу значительно упрощает разработку.

**Иерархическая** организация предполагает использование иерархий при разработке программной

системы. Иерархии упрощают систему абстракций. Иерархия - это упорядочивание абстракций,

расположение их по уровню.

Существуют два вида иерархий:

1. Часть - целое (Part-All). Предполагает, что некоторая абстракция включена в другую. Данный вид

иерархии используется в ранних этапах проектирования. На логическом уровне при разбиении

предметной области на объекты, на физическом уровне - при разбиении систем на модули. (связь

между объектами - агрегация)

2. Общее - частное (Is - A). Показывает, что некоторая абстракция является частным случаем другой.

Используется в основном механизмее ООП - наследовании.

**Наследование** - это отношение между классами, когда один из них использует структурную или

функциональную часть другого.

**Типизация** - ограничение, накладываемое на свойства объектов, препятствующее

взаимозаменяемости абстракций различного типа. Использование принципа типизации помогает

выполнить раннее обнаружение ошибок, упрощает комментирование, помогает генерировать более эффективный код

**Параллелизм** - свойство, которое позволяет нескольким абстракциям одновременно находиться в

активном состоянии. Данный принцип реализует ОС.

**Устойчивость (сохраняемость)** - свойство абстракции существовать во времени, независимо от

процесса, породившего ее. Различают следующие типы объектов:

1. Временные объекты

2. Локальные объекты

3. Глобальные объекты

4. Сохраняемые объекты (на внешних носителях)