**1 глава - понятие программной инженерии**

***Программная инженерия*** (промышленное программирование) ассоциируется с разработкой больших и сложных программ коллективами разработчиков.

**Становление и развитие этой области было вызвано рядом проблем:**

1) Высокая стоимость программного обеспечения.

2) Сложность его создания.

3) Необходимость управления и прогнозирования процессов разработки.

В конце 60-х/начале 70-х годов произошло событие, которое вошло в историю как ***первый кризис программирования***. Событие состояло в том, что стоимость программного обеспечения стала приближаться к стоимости железа, а динамика роста этой стоимости позволяла прогнозировать, что к середине 90-х годов всё человечество будет заниматься разработкой ПО.

С тех пор программная инженерия получила достаточно **бурное развитие**. Каждый этап развития связан с появлением или осознанием очередной проблемы и нахождением путей и способов решения этой проблемы.

Сам термин ***software engineering*** впервые был озвучен в 1968 году на конференции по науке и технике в Германии.

# **Предпосылки и история.**

## Повторное использование кода (модульное программирование).

На первых этапах становления программной инженерии было отмечено, что высокая стоимость программ связана с разработкой одинаковых и похожих фрагментов кода в различных программах. Повторное использование при создании новых программ ранее написанных фрагментов кода обещало существенное снижение сроков и стоимости разработки.

**Главный принцип модульного программирования** состоял в выделении таких фрагментов и оформлении их в виде отдельных модулей. Каждый модуль снабжался описанием, в котором устанавливались правила его использования - интерфейс модуля. Интерфейс задавал связи модуля с основной программой - связи по данным и связи по управлению. При этом возможность повторного использования модулей определялась количеством и сложностью этих связей. Наиболее простыми оказались модули решения математических задач.

Для многих других типов модулей возможность их повторного использования **оказалась проблематичной** ввиду сложности их связи с основной программой. Повторное использование модулей со сложными интерфейсами является достаточно актуальной и по сей день задачей.

## Рост сложности программ.

Следующий этап возрастания стоимости ПО был связан с переходом от разработки относительно простых программ к разработке сложных программных комплексов. Следует отметить, что этот переход был вызван появлением вычислительной техники 3-го поколения. С переходом на использование интегральных схем производительность компьютеров возросла на порядки, что и создало предпосылки для решения различных сложных задач.

**К сложным задачам относятся:**

* Система управления космическими объектами
* Управление оборонным комплексом
* Автоматизация технологических процессов на производстве
* Автоматизация финансовых учреждений
* И так далее…

**Сложность можно оценивать:**

1. Большой объём кода
2. Большое количество связей между элементами кода
3. Большое количество разработчиков
4. Большое количество пользователей
5. Длительное время использования

Для таких сложных задач оказалось, что основная часть их стоимости приходится не на создание программ, а на их внедрение и эксплуатацию.

По аналогии с промышленной технологией стали говорить о жизненном цикле программного продукта, как о последовательности определённых этапов:

1. Эта проектирования
2. Этап разработки
3. Этап тестирования
4. Этап сопровождения

## Основные принципе и технологии структурного проектирования и кодирования.

1. Нисходящее функциональное проектирование
2. Применение специальных языков проектирования и средств автоматизация использования этих языков
3. Дисциплина проектирования и разработки

## Модификация программы

Следующая проблема роста стоимости программ была связана с тем, что изменения требований к программе стали возникать не только на этапе сопровождения, но и на этапе проектирования.

Создание программного продукта превратилось в его перманентное проектирование. Возник вопрос: «Как проектировать и писать программы, чтобы обеспечить возможность внесения изменений в программу, не меняя ранее написанного кода?».

## Объектно-ориентированное программирование (ООП)

**ДОПИСАТЬ (В ТГ)**

## Итоги

***Программная инженерия*** (технология промышленного программирования) как направление возникло и формировалось под давлением роста стоимости создаваемого программного обеспечения. Главная цель этой области знаний – сокращение стоимости и сроков разработки программы.

Программная инженерия прошла ***несколько этапов развития***, в процессе которых были сформулированы фундаментальные принципы и методы разработки программных продуктов. Основной принцип программной инженерии состоит в том, что программы создаются в результате выполнения нескольких взаимосвязанных этапов (анализ требований, проектирование, разработка, тестирование), составляющих жизненный цикл программного продукта.

## Продолжение кризиса программирования

Несмотря на то, что программная инженерия достигла определённых успехов, **перманентный кризис программирования продолжается**. Связано это с тем, что на рубеже 80-х – 90-х годов началась **информационно-техническая революция**, вызванная взрывным ростом использования информационных средств.

* **Успешный проект** – вовремя и в рамках бюджета был выполнен весь намеченный фронт работы.
* **Проблемный проект** – нарушение сроков, перерасход бюджета или реализация не полноценная
* **Провальный проект** – большой перерасход бюджета

## Понятие программной инженерии

На сегодняшний день нет единого определения понятия «***программная инженерия*».**

Существует несколько трактовок:

1. Установление и использование обоснованных инженерных принципов для экономного получения ПО, которое надёжно и работает на реальных машинах
2. Та форма инженерии, которая применяет принципы информатики ***(computer science)*** и математики для рентабельного ращения проблем через ПО
3. Применении систематического, дисциплинированного и измеряемого подхода к разработке, использованию и сопровождению.
4. Дисциплина, целью которой является создание качественного ПО, создание которого завершается вовремя, не превышает выделенных бюджетных средств и удовлетворяет выдвигаемым требованиям.