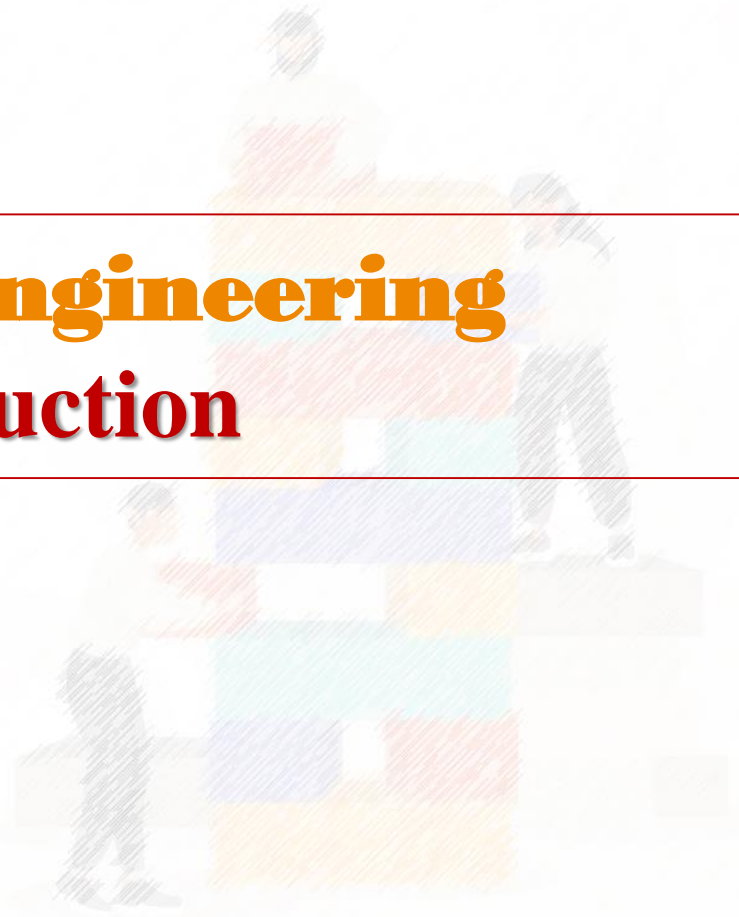


# **Software Engineering**

## **Introduction**



# Course Objective

- Understand the meaning of Software Engineering
- Understand software development process
- Understand Software Engineers and Software programmers, QA and QC,...

# What is software?

- Computer **programs** and associated **documentation**.
- Software products may be developed for
  - a particular customer or
  - a general market

# Software products

- Generic products
  - Stand-alone systems that are marketed and sold to any customer who wishes to buy them.
  - Examples ???
- Customized products
  - Software that is commissioned by a specific customer to meet their own needs.
  - Examples ???

# What is software engineering?

- All aspects of software production
  - *Not just technical* process of development. Also **project management** and the development of **tools**, **methods** etc. to support software production.
- Software engineering is an engineering **discipline** that is concerned with all aspects of software production
  - from the early stages of system **specification** through
  - to maintaining the system after it has gone into use.

# Software process activities

## 1. Software **Specification/Requirement**

- where customers and engineers define the software that is to be produced and the constraints on its operation.

## 2. Software **Design**

- where the software is designed

## 3. Software **Implementation**

- where the software is programmed

## 4. Software **Testing/Validation**

- where the software is checked to ensure that it is what the customer requires.

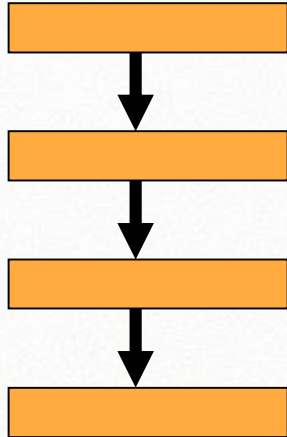
## ● **Software Evolution/Maintenance**

- where the software is modified to reflect changing customer and market requirements



# Forward Engineering vs Reverse Engineering

Forward Engineering



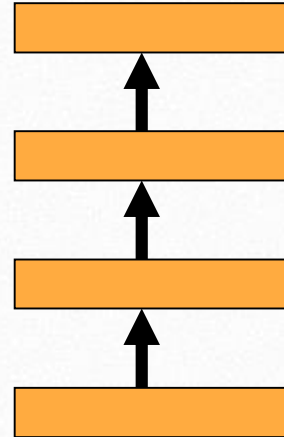
What to do

How to do it

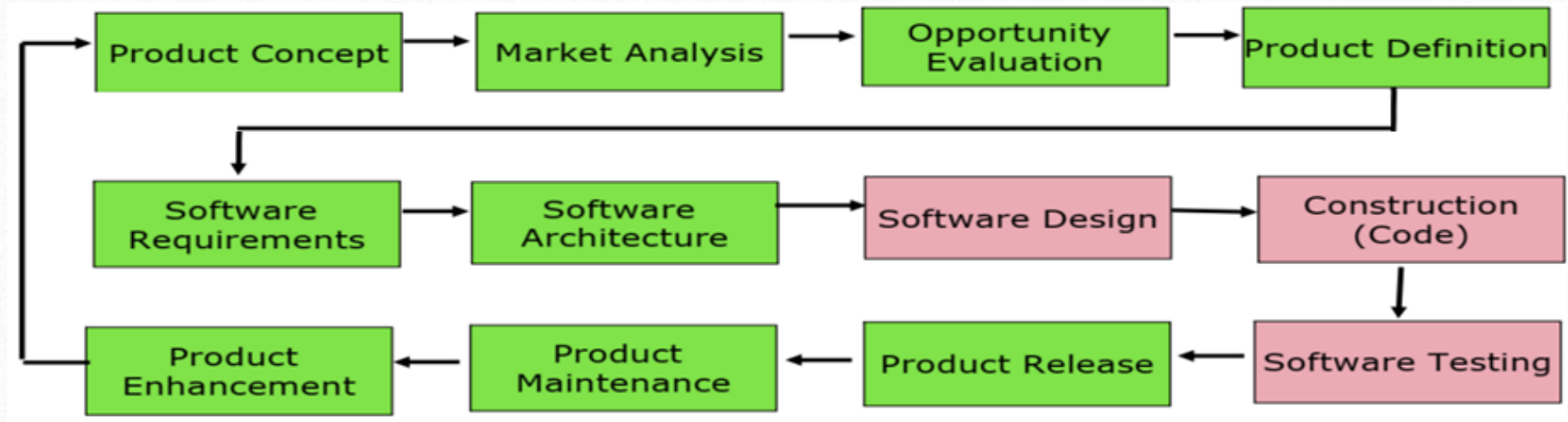
Do it

Test it and Use it

Reverse Engineering



# Example: Product Development Process





# Software Engineer vs. Programmer

- The differences of Software Engineer and Programmer?

# Some Comparisons

Issue	Software Engineer	Programmer
Scope	The entire project from requirements to final products	Coding, some design and tests
Context	Teamwork, collaborate with all stakeholders	Individual work, coordinate with team members
Focus	Software product & application for business solution	Project focus
Skills	Domain knowledge	Mathematics and programming

# Computer Science vs. Software Engineering

- The differences of Computer Science and Software Engineering?

# Computer Science Focus

- Mathematical logic
- Algorithms
- Data structures
- Compilers
- Interpreters
- Programming languages

# Software Engineering Focus

- Requirements Engineering
- Software architecture
- Software design
- Software construction
- Software validation
- Software configuration management
- Software quality assurance
- Software project management

# Quality assurance vs. Quality control



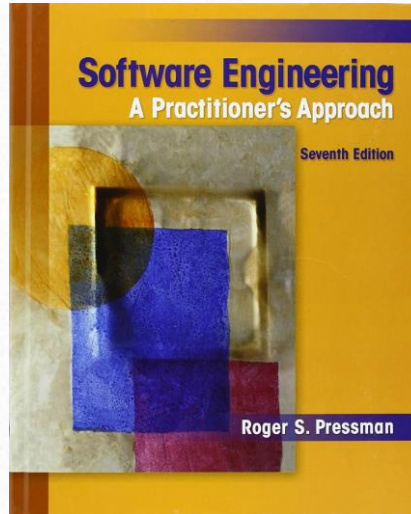
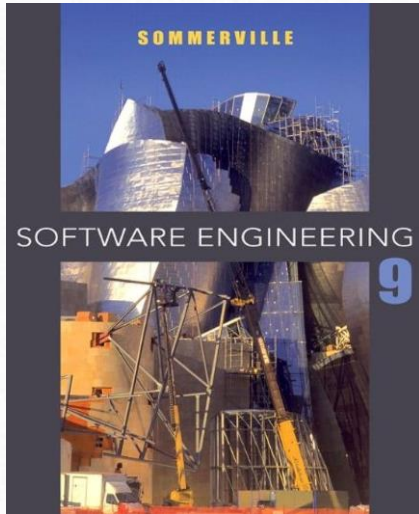


# Why Study Software Engineering

- It is easy to find people with programming skills in C++, Java, .Net,... but difficult to find people with the knowledge necessary for engineering large software systems.
- The history of software development is one of increasing scale and complexity
- Today large software projects require many teams of hundreds of people, with the complexity outpacing the ability to intuitively solve problems
- What is required is a more structured and disciplined approach to software development and management... and the world needs: Software Engineering

# References

- [1] Software Engineering, Ian Sommerville, 9<sup>th</sup> Edition, Addison-Wesley 2011
- [2] Software Engineering, Roger S. Pressman, 7<sup>th</sup> Edition, 2010



# Assessment

- Bài tập tại lớp: 30%
- Seminar (có thuyết trình): 20%
- Tiểu luận cuối kỳ: 50%