

# CÔNG NGHỆ PHẦN MỀM (Software Engineering)

GV : Ths. Nguyễn Thị Thanh Trúc

Email: [trucntt@uit.edu.vn](mailto:trucntt@uit.edu.vn)

# GIỚI THIỆU CHUNG

## ❖ Mục tiêu môn học

- ✓ Cung cấp cho sinh viên các kiến thức và kỹ năng làm phần mềm một cách chuyên nghiệp, qui trình phát triển phần mềm.
- ✓ Định hướng cho việc tiếp cận các kiến thức liên quan

## ❖ Số đơn vị học trình

- ✓ 3 tín chỉ LT + (1 tín chỉ TH)

## ❖ Môn học trước

- ✓ Nhập môn Cơ sở dữ liệu

# NỘI DUNG MÔN HỌC

- Chương 1: Tổng quan phần mềm và công nghệ phần mềm
- Chương 2: Quy trình và mô hình quy trình phần mềm
- Chương 3: Phân tích và đặc tả yêu cầu
- Chương 4: Thiết kế phần mềm
- Chương 5: Lập trình
- Chương 6: Kiểm thử
- Chương 7: Tích hợp, chuyển giao và bảo trì

# NỘI DUNG MÔN HỌC

- Tổng quan phần mềm và công nghệ phần mềm
  - Vai trò và tầm quan trọng của phần mềm
  - Khó khăn và thách thức với phát triển phần mềm
  - Tổng quan về công nghệ phần mềm
- Quy trình và mô hình quy trình phần mềm
  - Khái niệm
  - Các mô hình quy trình tổng quát, cụ thể
- Phân tích và đặc tả yêu cầu
  - Vai trò của phân tích và đặc tả yêu cầu
  - Các hoạt động phân tích, đặc tả yêu cầu
  - Phương pháp, công cụ để phân tích, đặc tả yêu cầu

# NỘI DUNG MÔN HỌC

## ■ Thiết kế phần mềm

- ❑ Khái niệm, nguyên lý, chất lượng thiết kế phần mềm
- ❑ Thiết kế kiến trúc, thiết kế giao diện
- ❑ Một số phương pháp, công cụ thiết kế

## ■ Lập trình

- ❑ Phong cách lập trình
- ❑ Lập trình tránh lỗi
- ❑ Lập trình hiệu quả

# Nội dung môn học

## ■ Kiểm thử

- Khái niệm, các loại kiểm thử
- Quy trình, công cụ trợ giúp kiểm thử
- Viết tài liệu kiểm thử

## ■ Tích hợp, chuyển giao và bảo trì

- Khái niệm
- Các nguyên lý, hoạt động tích hợp
- Các hoạt động chuyển giao, bảo trì
- Các vấn đề của hoạt động bảo trì

# Thực hành

Thực hiện đồ án môn học

- ❖ Xây dựng mô hình dữ liệu ERD dùng PowerDesigner
- ❖ Quản lý CSDL với SQL Server, MySQL
- ❖ Ngôn ngữ lập trình tùy chọn: C++, C# hoặc Java

# Hình thức kiểm tra & đánh giá

- ❖ Thi lý thuyết: 50% Tự luận
- ❖ Thực Hành: 50%
  - ❖ Báo cáo seminar: 20% (Giữa kỳ)
  - ❖ Thu hoạch môn học: 30% (phát triển từ seminar)



# TÀI LIỆU THAM KHẢO

- 1. Ian Sommerville, Le Génie Logiciel, Addison wesley, 2004, 7<sup>th</sup> Edition
- 2. Ivar Jacobson, Object – Oriented Software Engineering, Addison Wesley, 1992
- 3. James Rumbaugh, O.O. Modeling and Design. Prentice Hall International, 1991
- 4. Roger S. Pressman, SE A Practioner's Approach, Mc Graw Hill International, 2005.
- <http://www.cs.st-andrews.ac.uk/~ifs/Books/SE7/index.html>
- Tài liệu bài giảng môn học
  - [Bài giảng trên course.uit.edu.vn](http://course.uit.edu.vn)

# CHƯƠNG 1

# TỔNG QUAN CÔNG NGHỆ

# PHẦN MỀM

# TỔNG QUAN VỀ CÔNG NGHỆ PHẦN MỀM

- Khái niệm chung
- Tầm quan trọng của phần mềm
- Đặc trưng của phần mềm
- Các loại phần mềm và đặc điểm của mỗi loại phần mềm
- Tiến hoá của phần mềm
- Tiêu chí cho phần mềm tốt
- Công nghệ phần mềm

# KHÁI NIỆM VỀ PHẦN MỀM

- Ví dụ: xét một số phần mềm sau
  - ❑ Phần mềm quản lý học sinh cấp 3.
  - ❑ Phần mềm quản lý thư viện.
  - ❑ Phần mềm quản lý nhà sách.
  - ❑ Phần mềm quản lý khách sạn.
  - ❑ Phần mềm quản lý phòng mạch tư.
  - ❑ Phần mềm quản lý sổ tiết kiệm.
  - ❑ Phần mềm quản lý bán vé chuyến bay.
  - ❑ Phần mềm xếp thời khóa biểu

# KHÁI NIỆM VỀ PHẦN MỀM

- Phần mềm là gì?
- Được xem xét ở hai góc độ:
  - Góc nhìn người sử dụng
  - Góc nhìn chuyên viên tin học

# KHÁI NIỆM VỀ PHẦN MỀM

- Phần mềm dưới **Góc nhìn người sử dụng**
  - **Chương trình** thực thi được trên máy tính hoặc các thiết bị chuyên dụng khác
  - Nhằm **hỗ trợ** cho các nhà chuyên môn trong từng lĩnh vực chuyên ngành thực hiện **tốt hơn** các thao tác nghiệp vụ của mình

# KHÁI NIỆM VỀ PHẦN MỀM

- Môi trường triển khai phần mềm
  - Máy tính: Desktop, Laptop, Tablet PC, ...
  - Thiết bị chuyên dụng:
    - Thiết bị di động: PDA, Pocket PC, ĐTDĐ
    - Các thiết bị chuyên dụng khác: set-top box, router
- Hỗ trợ làm tốt hơn các thao tác nghiệp vụ:
  - Tin học hóa nghiệp vụ hiện đang làm thủ công
  - Cải tiến chức năng nghiệp vụ hiện đang được thực hiện trên máy tính
  - Đề ra, xây dựng và triển khai chức năng nghiệp vụ mới

# KHÁI NIỆM VỀ PHẦN MỀM

- Phần mềm dưới góc nhìn của chuyên viên Tin học
  - Đây là một hệ thống bao gồm 3 thành phần cơ bản:
    - Thành phần giao tiếp
    - Thành phần xử lý
    - Thành phần lưu trữ
  - ➔ Cần được xây dựng để thực hiện theo yêu cầu của người sử dụng



# KHÁI NIỆM VỀ PHẦN MỀM

## ■ Thành phần giao tiếp (Giao diện)

- ❑ Cho phép **nhập/xuất thông tin** cùng với hình thức trình bày/giao tiếp tương ứng
- ❑ Nhằm đưa thông tin từ thế giới thực bên ngoài phần mềm (người sử dụng, các thiết bị, phần mềm khác...) vào bên trong, hoặc ngược lại
- ❑ Cho phép **tiếp nhận** các yêu cầu về việc sử dụng phần mềm từ người sử dụng, từ các thiết bị thu thập dữ liệu, hoặc từ các phần mềm khác
- ❑ Cho phép **trình bày** các kết quả của việc thực hiện các yêu cầu cho người dùng hoặc điều khiển hoạt động các thiết bị điều khiển

# KHÁI NIỆM VỀ PHẦN MỀM

## ■ Thành phần xử lý:

- ❑ **Kiểm tra tính hợp lệ** của thông tin nguồn được cung cấp từ người dùng theo các quy định ràng buộc trong thế giới thực
- ❑ **Tiến hành xử lý** cho ra kết quả mong đợi theo *quy định* tính toán có sẵn *trong thế giới thực*
- ❑ Hoặc **tiến hành xử lý** theo *thuật giải tự đề xuất*
- ❑ Việc xử lý dựa trên thông tin nguồn từ **người sử dụng cung cấp**
- ❑ Việc xử lý cho ra **kết quả** có thể dùng để *xuất cho người dùng xem qua thành phần giao diện*, hay *lưu trữ lại qua thành phần lưu trữ*, hoặc cả hai

# KHÁI NIỆM VỀ PHẦN MỀM

- Thành phần xử lý (tt)
  - Thành phần xử lý là hệ thống chuyên xử lý tính toán, biến đổi dữ liệu
    - Dùng thông tin nguồn từ thành phần giao diện (**chức năng nhập**) hay thành phần dữ liệu (**chức năng đọc**)
    - Kiểm tra tính hợp lệ (chức năng kiểm tra) và sau đó tiến hành xử lý (**chức năng xử lý**) – nếu cần thiết
    - Cho ra kết quả sẽ được trình bày thông qua thành phần giao diện (**chức năng xuất**) hoặc lưu trữ lại trong thành phần dữ liệu (**chức năng ghi**)

# KHÁI NIỆM VỀ PHẦN MỀM

- Thành phần lưu trữ (thành phần dữ liệu)
  - Cho phép lưu trữ lại (**chức năng ghi**) các kết quả đã xử lý
  - Cho phép truy xuất lại (**chức năng đọc**) các dữ liệu đã lưu trữ phục vụ cho các hàm xử lý tương ứng
- ➔ thành phần dữ liệu là hệ thống chuyên **đọc ghi** dữ liệu cùng với mô hình tổ chức lưu trữ dữ liệu tương ứng
- Mục tiêu chính của thành phần này là **chuyển đổi dữ liệu** giữa bộ nhớ chính và bộ nhớ phụ

# Khái niệm về lớp phần mềm

- Phân lớp theo mục đích sử dụng:
  - Lớp phần mềm quản lý học sinh
  - Lớp phần mềm đánh cờ
  - Lớp phần mềm giải bài tập
  - Phần mềm soạn thảo văn bản
  - ...

# Khái niệm về lớp phần mềm

- Phân lớp theo mục tiêu tạo lập
  - Lớp phần mềm theo hợp đồng
  - Lớp phần mềm khung
  - Lớp phần mềm đóng gói

# Khái niệm về lớp phần mềm

## ■ Lớp phần mềm

- Lớp phần mềm là hệ thống các phần mềm trên cùng một lĩnh vực hoạt động nào đó
- Do cùng lĩnh vực hoạt động nên các phần mềm cùng lớp thường có cấu trúc và chức năng tương tự nhau

# ĐỊNH NGHĨA PHẦN MỀM

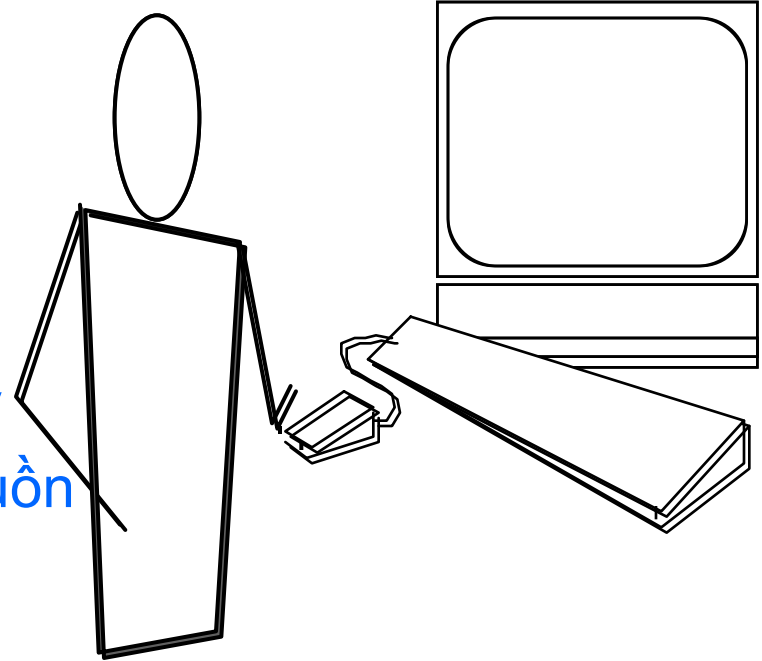
## ■ Phần mềm bao gồm

### □ Chương trình máy tính

- Phần vận hành được: mã máy
- Phần không vận hành: mã nguồn

### □ Tài liệu đi kèm chương trình

- Tài liệu về thông tin cài đặt
- Tài liệu hướng dẫn sử dụng
- Tài liệu kỹ thuật: tài liệu đặc tả, thiết kế phần mềm





# VAI TRÒ CỦA PHẦN MỀM

- Phần mềm đóng vai trò đặc biệt quan trọng trong xã hội hiện đại
- Vai trò của phần mềm xét trên các mặt:
  - Phạm vi ứng dụng
  - Ảnh hưởng tới hệ thống
  - Ảnh hưởng tới tổ chức
  - Ảnh hưởng kinh tế quốc gia

# VAI TRÒ CỦA PHẦN MỀM

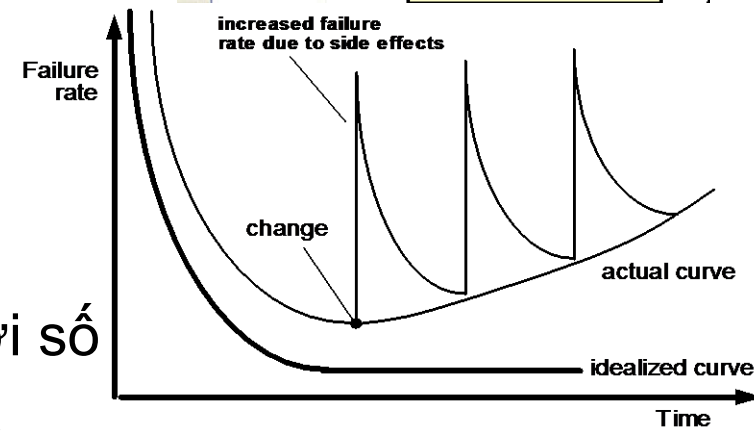
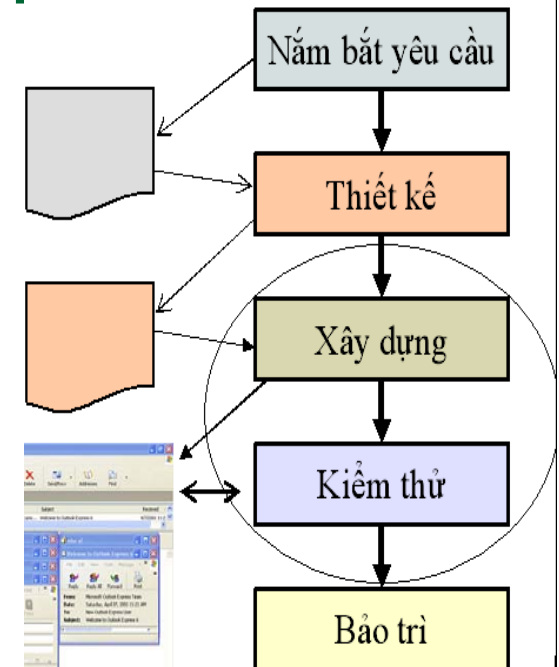
- Phần mềm được ứng dụng trên mọi lĩnh vực của xã hội
- Ví dụ
  - Y tế: Phần mềm đo điện tâm đồ của người bệnh
  - Quân sự: Phần mềm mô phỏng đường bay của đạn
  - Tài chính: Phần mềm kế toán
  - Giáo dục và đào tạo: Phần mềm quản lý kết quả học tập

# VAI TRÒ CỦA PHẦN MỀM

- Ngày càng nhiều hệ thống được phần mềm điều khiển, trợ giúp
  - Tính tự động của hệ thống ngày một cao
  - Chi phí phần mềm >> phần cứng
- Phần mềm tạo nên sự khác biệt của tổ chức
  - Thay đổi phong cách làm việc
  - Quyết định năng suất của tổ chức
- Nền kinh tế mọi nước đều phụ thuộc vào phần mềm
  - Phần mềm sai hỏng, kinh tế tổn thất
  - Chi phí cho phần mềm chiếm tỷ trọng đáng kể trong GNP ở các nước phát triển

# ĐẶC TRƯNG CỦA PHẦN MỀM

- Sản xuất mang tính thủ công
- Khó kiểm soát chất lượng ở khâu trung gian
- Khó dự đoán trước về hiệu năng
- Được đặt hàng hoàn chỉnh theo từng yêu cầu riêng
- Thoái hoá theo thời gian
  - Môi trường sử dụng
  - Nhu cầu thay đổi
  - Lỗi sinh ra do nâng cấp
- Phần mềm được phát triển theo nhóm
  - Năng lực nhóm không tuyến tính với số lượng thành viên
  - Thời gian cho trao đổi thông tin chiếm tỷ lệ cao



# PHÂN LOẠI PHẦN MỀM

- Trên quan điểm của người phát triển có 2 loại
  - 1. Sản phẩm đặt hàng
    - Sản xuất theo đơn đặt hàng (vd: HTTT quản lý)
    - Đơn chiếc, yêu cầu đặc thù
  - 2. Sản phẩm chung
    - Bán rộng rãi
    - Thoả mãn yêu cầu chung của số lớn người

# PHÂN LOẠI PHẦN MỀM

## ■ Theo lĩnh vực ứng dụng:

### □ 1. Phần mềm hệ thống

- Tập hợp các chương trình
- Tương tác trực tiếp với phần cứng
- Phục vụ nhiều người dùng
- Ví dụ: Hệ điều hành Window 9x, Unix

### □ 2. Phần mềm thời gian thực

- Thu thập xử lý các dữ liệu thế giới thực
- Đáp ứng yêu cầu chặt chẽ về thời gian
  - Thu thập dữ liệu
  - Phân tích dữ liệu
  - Kiểm soát, điều khiển
  - Điều phối

Ví dụ phần mềm nghe nhạc Window Media

# PHÂN LOẠI PHẦN MỀM

## ■ 3. Phần mềm nghiệp vụ

- ❑ Xử lý thông tin nghiệp vụ, thường gắn với CSDL
- ❑ Xử lý các giao tác
- ❑ Lĩnh vực ứng dụng rất lớn
- ❑ Ví dụ phần mềm kế toán

## ■ 4. Phần mềm khoa học kỹ thuật

- ❑ Đặc trưng bởi thuật toán (tính toán vật lý, mô phỏng)
- ❑ Đòi hỏi năng lực tính toán cao
- ❑ Ví dụ: phần mềm hỗ trợ dự báo thời tiết

## ■ 5. Phần mềm nhúng (embed software)

- ❑ Chỉ đọc khi thiết bị khởi động
- ❑ Thực hiện chức năng hạn chế (điều khiển sản phẩm)
- ❑ Là sự kết hợp giữa hệ thống và thời gian thực
- ❑ Ví dụ: Phần mềm nghe nhạc được nhúng trong đầu máy DVD

# PHÂN LOẠI PHẦN MỀM

## ■ 6. Phần mềm máy tính cá nhân

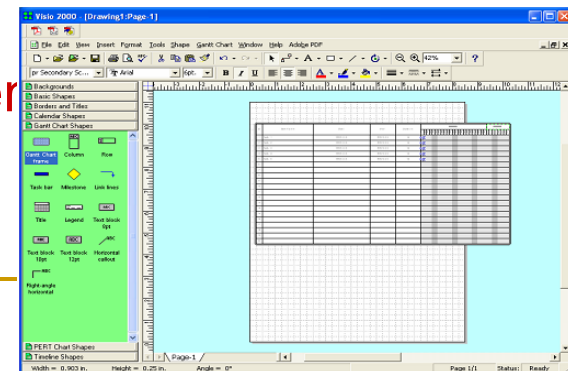
- ❑ Các bài toán nghiệp vụ nhỏ (ứng dụng văn phòng)
- ❑ Giao diện đồ họa phát triển
- ❑ Có nhu cầu rất cao
- ❑ Ví dụ
  - Phần mềm Office, Phần mềm nén file Winzip

## ■ 7. Phần mềm công cụ cho kỹ nghệ phần mềm

- ❑ Dùng thuật toán phi số
- ❑ Hệ chuyên gia, nhận dạng, trò chơi
- ❑ Ví dụ:
  - phần mềm nhận dạng tiếng nói,
  - Phần mềm chơi cờ

## ■ 8. Phần mềm công cụ cho công nghệ phần mềm

- ❑ Compiler, các công cụ CASE
- ❑ Ví dụ: Visio 2000, Rational Rose





# TIẾN HOÁ CỦA PHẦN MỀM

- ~1960:
  - ❑ Xử lý theo lô
  - ❑ Không có tài liệu tường minh
  - ❑ \* Ngôn ngữ (NN) bậc thấp, Không có PP
- ~1970:
  - ❑ Hệ thống đa nhiệm, đa người dùng
  - ❑ Hệ thống thời gian thực
  - ❑ \* NN bậc cao, PP có cấu trúc
- Gần đây:
  - ❑ Hệ thống phân tán
  - ❑ Tăng qui mô và độ phức tạp
  - ❑ \* NN bậc cao, PP hướng đối tượng
  - ❑ Phần cứng ngày càng phát triển
  - ❑ Hệ thống phần mềm ngày càng đa dạng, phong phú, xử lý càng phức tạp, công nghệ ngày càng phát triển

# TIẾN HOÁ CỦA PHẦN MỀM

- Phần mềm ngày càng phức tạp
  - Unix: 4M dòng lệnh
  - Window2K: ~100M dòng lệnh
- Lý do:
  - Năng lực máy tính ngày càng mạnh
  - Các hệ thống máy tính được liên kết lại
  - Ham muốn của người dùng ngày càng lớn
- Phần mềm trở thành ngành công nghiệp

# TIÊU CHÍ PHẦN MỀM TỐT

## ■ 1. Tính bảo trì được

- Dễ sửa lỗi
- Dễ nâng cấp
- ➔ - Thiết kế dễ hiểu (thiết kế tốt)
  - Cài đặt bằng ngôn ngữ cấp cao
  - Có đầy đủ tài liệu

## ■ 2. Tính tin cậy

- Phần mềm không có khiếm khuyết
- Đáp ứng được nhu cầu người sử dụng
- ➔ Phát hiện và sửa:
  - Lỗi lập trình
  - Lỗi phân tích, thiết kế
    - Sai, thiếu chức năng
    - Hoạt động không hiệu quả

# TIÊU CHÍ PHẦN MỀM TỐT

## ■ 3. Tính hiệu quả

- Khi vận hành không lãng phí tài nguyên phần cứng
  - Bộ nhớ trong
  - Tốc độ
  - Bộ nhớ ngoài
  - → **Dùng ngôn ngữ bậc thấp**  
**Truy cập trực tiếp đến thiết bị**

## ■ 4. Tính dễ dùng

- Giao diện nhất quán, phù hợp với trình độ của người dùng
- Có tài liệu, tiện ích trợ giúp

## ■ Nhận xét

- Khó tối ưu đồng thời các thuộc tính
  - **Hiệu quả >< dễ bảo trì, dễ sử dụng**
- Chi phí cho tối ưu là không tuyến tính: giá thành, hiệu năng

# TỔNG KẾT

- Phần mềm có tầm quan trọng đặc biệt: tạo nên sự khác biệt của tổ chức, của hệ thống
- Phát triển phần mềm là công việc khó khăn, phức tạp, rủi ro
  - Là phần tử logic, không có độ đo trực quan, khó kiểm soát chất lượng khi phát triển
  - Không được định hình trước, khó dự đoán hiệu năng khi chưa có sản phẩm
  - Mang tính thủ công, phụ thuộc vào con người
  - Bị ảnh hưởng lớn từ môi trường => nhiều rủi ro
  - Yêu cầu cao cho phần mềm tốt

# PHẦN MỀM VÀ CÔNG NGHỆ PHẦN MỀM

- Khái niệm
- Các yếu tố cấu thành
- Các giai đoạn phát triển
- Các hoạt động

# Software Engineering

## ■ Classic Definition (1969)

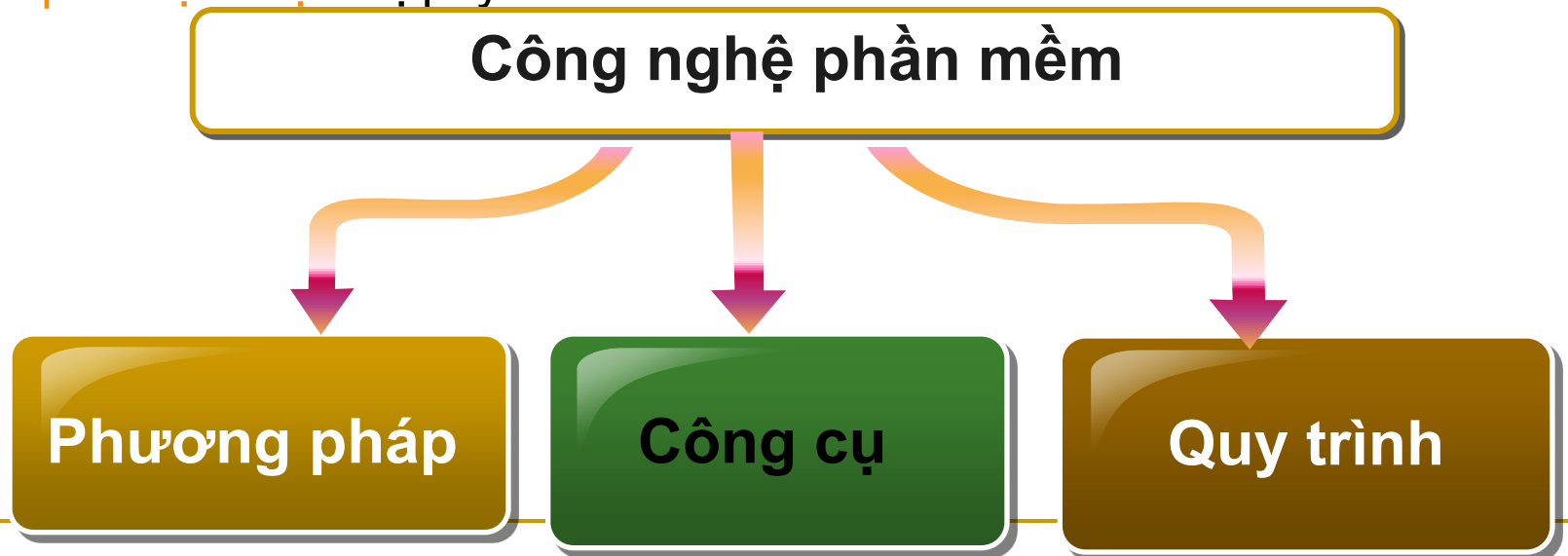
*“The establishment and use of sound engineering principles in order to obtain economically software that is reliable and works efficiently on real machines.”*

## ■ IEEE Definition (1993)

*“Software Engineering: (1) The application of a systematic, disciplines, quantifiable approach to the development, operation, and maintenance of software; that is the application of engineering to software. (2) The study of approaches as in (1).”*

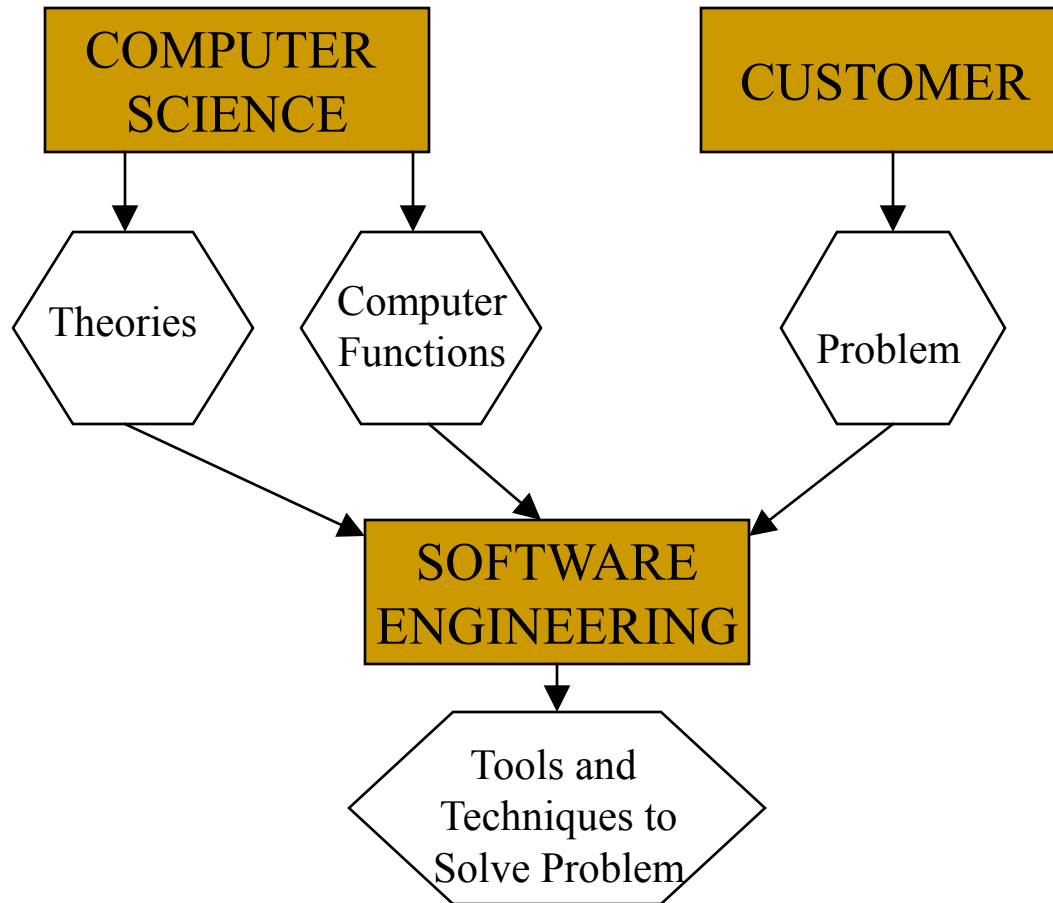
# KHÁI NIỆM

- Công nghệ phần mềm: software-engineering
  - Công nghệ phần mềm là bộ môn tích hợp cả quy trình, phương pháp, công cụ hỗ trợ cho phát triển phần mềm máy tính [Pressman 1995]
  - **Công nghệ phần mềm** là ngành khoa học nghiên cứu về việc xây dựng các phần mềm **có chất lượng cao** trong **thời gian** và **chi phí thực hiện** hợp lý





# Problem Solving



# KHÁI NIỆM

- Chức năng của công nghệ phần mềm
  - Thiết lập và áp dụng các nguyên lý công nghệ cho phát triển phần mềm máy tính
    - Phương pháp
    - Công cụ
    - Quy trình
    - Nhiều phiên bản
    - Nhiều người tham gia
    - Được lượng hoá
  - ➔ Phần mềm tin cậy, hiệu quả trong giới hạn

# CÁC YẾU TỐ CỦA SE

- 3 yếu tố chính
  - Thủ tục (procedures)
  - Phương pháp (methods)
  - Công cụ (tools)

# CÁC YẾU TỐ CỦA SE

## ■ Phương pháp:

- Hệ thống các hướng dẫn cho phép từng bước thực hiện một giai đoạn nào đó trong quy trình phần mềm
- Cách làm cụ thể để xây dựng phần mềm
- Mỗi công đoạn làm phần mềm có phương pháp riêng
  - Phương pháp Phân tích
  - Phương pháp Thiết kế
  - Phương pháp Lập trình
  - Phương pháp Kiểm thử
- Phương pháp có cấu trúc (DeMarco, 1978)
- Phương pháp hướng đối tượng (Booch 1994, Rumbaugh 1991)

# CÁC YẾU TỐ CỦA SE

## ■ Công cụ:

### □ Cung cấp sự trợ giúp tự động/bán tự động cho từng phương pháp

- Hỗ trợ quản lý dự án: **Microsoft Project**
- Hỗ trợ phân tích, thiết kế: **Power Designer**
- Hỗ trợ viết chương trình, biên dịch, gỡ rối: **Jbuilder**,
- Hỗ trợ kiểm thử: **Defect Management System**

### □ **CASE Tools** (Computer Aided Software Engineering Tools)

- Hệ thống các phần mềm trợ giúp trong lĩnh vực xây dựng phần mềm
- Hỗ trợ các chuyên viên tin học trong các bước xây dựng phần mềm theo một phương pháp nào đó với một quy trình được chọn trước

# CÁC YẾU TỐ CỦA SE

## ■ Thủ tục (qui trình):

- Hệ thống các giai đoạn mà quá trình phát triển phần mềm phải trải qua
  - Xác định trình tự thực hiện công việc
  - Xác định các tài liệu, sản phẩm cần bàn giao và cách thức thực hiện
  - Định các mốc thời gian (milestones) và sản phẩm bàn giao
-

# CÁC GIAI ĐOẠN PHÁT TRIỂN

## ■ ~1970:

- ❑ PP lập trình có cấu trúc
- ❑ PP lập trình Top-down
- ❑ PP thiết kế chương trình theo mô đun

-NN bậc cao  
-PP có cấu trúc

## ■ ~1980:

- ❑ PP phân tích, thiết kế hệ thống hướng cấu trúc
- ❑ CC hỗ trợ phân tích, thiết kế, lập trình
- ❑ Độ đo phần mềm

-NN bậc cao  
-PP hướng đối tượng

## ■ Gần đây:

- ❑ Công cụ hướng đối tượng
- ❑ CASE-Tools hoàn thiện
- ❑ Phát triển các mô hình quản lý

# 4 CHỮ P

**Con người**



People

Người thực hiện

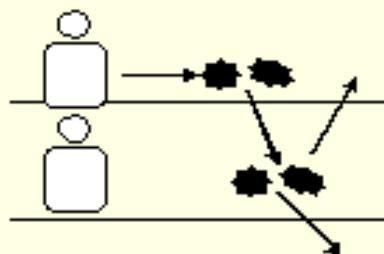
**Tiến trình**



Process

Cách thực hiện

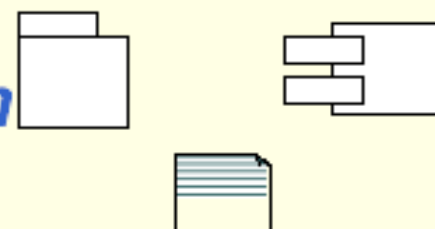
**Dự án**



Project

Việc/quá trình thực hiện

**Sản phẩm**



Product

Kết quả



# CÁC HOẠT ĐỘNG CỦA SE

- Xác định **qui trình** phát triển phần mềm được sử dụng
- Quản lý **dự án** phát triển phần mềm
- Mô tả về **sản phẩm** phần mềm được mong đợi
  - Nắm bắt và phân tích yêu cầu
- Thiết kế **sản phẩm**
- Cài đặt **sản phẩm**
- Kiểm thử thành phần của **sản phẩm**
- Tích hợp các thành phần của sản phẩm và kiểm thử kết quả tích hợp
- Bảo trì **sản phẩm**

# Mục tiêu của ngành Công nghệ Phần mềm

- ❑ Xây dựng được phần mềm **có chất lượng**
- ❑ Dễ dàng **xây dựng phần mềm mới** từ các phần mềm **có sẵn cùng lớp**

# Chất lượng phần mềm

Tính đúng đắn:

- Đầy đủ
- Chính xác

Tính đúng đắn

Phần mềm

Con người

Phần mềm khác

Phần cứng

Dưới góc nhìn của Người sử dụng

# Chất lượng phần mềm

Tính tiện dụng:

- Dễ học
- Dễ sử dụng
- Giao diện trực quan
- Tự nhiên

Tính tiện dụng



Con người

Tính đúng đắn



Phần mềm



Phần mềm khác



Phần cứng

Dưới góc nhìn của Người sử dụng

# Chất lượng phần mềm

Tính hiệu quả:

- Tối ưu sử dụng CPU\*
- Tối ưu sử dụng bộ nhớ\*
- Tối ưu sử dụng thiết bị

Tính đúng đắn

Tính tiện dụng



Con người



Phần mềm



Tính hiệu quả



Phần cứng



Phần mềm khác



Dưới góc nhìn của Người sử dụng

# Chất lượng phần mềm

Tính tương thích:

- Import/Export dữ liệu
- Tương tác

Tính đúng đắn

Tính tiện dụng

Tính hiệu quả



Con người



Phần mềm



Phần cứng



Phần mềm khác

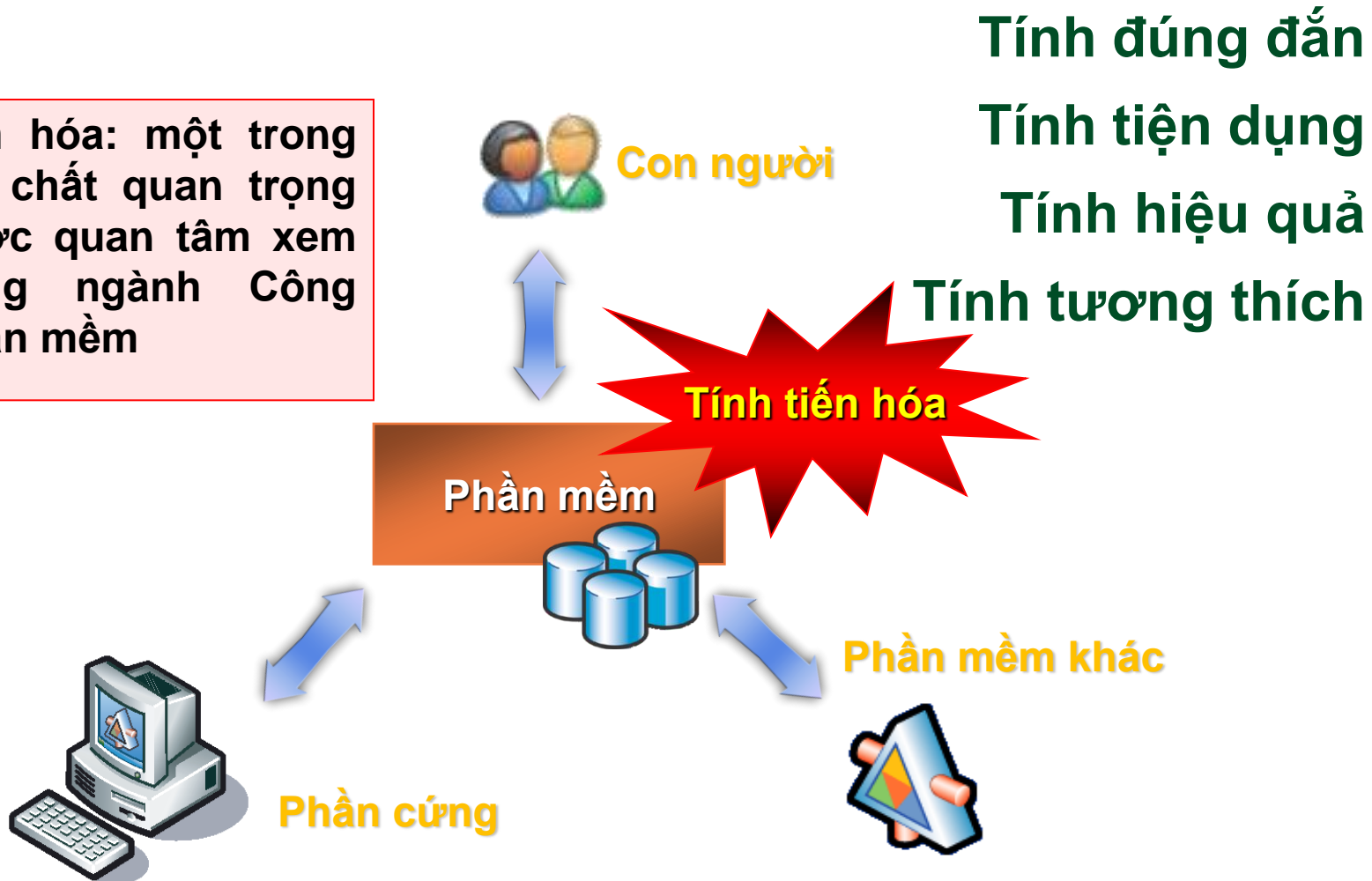


Tính tương thích

Dưới góc nhìn của Người sử dụng

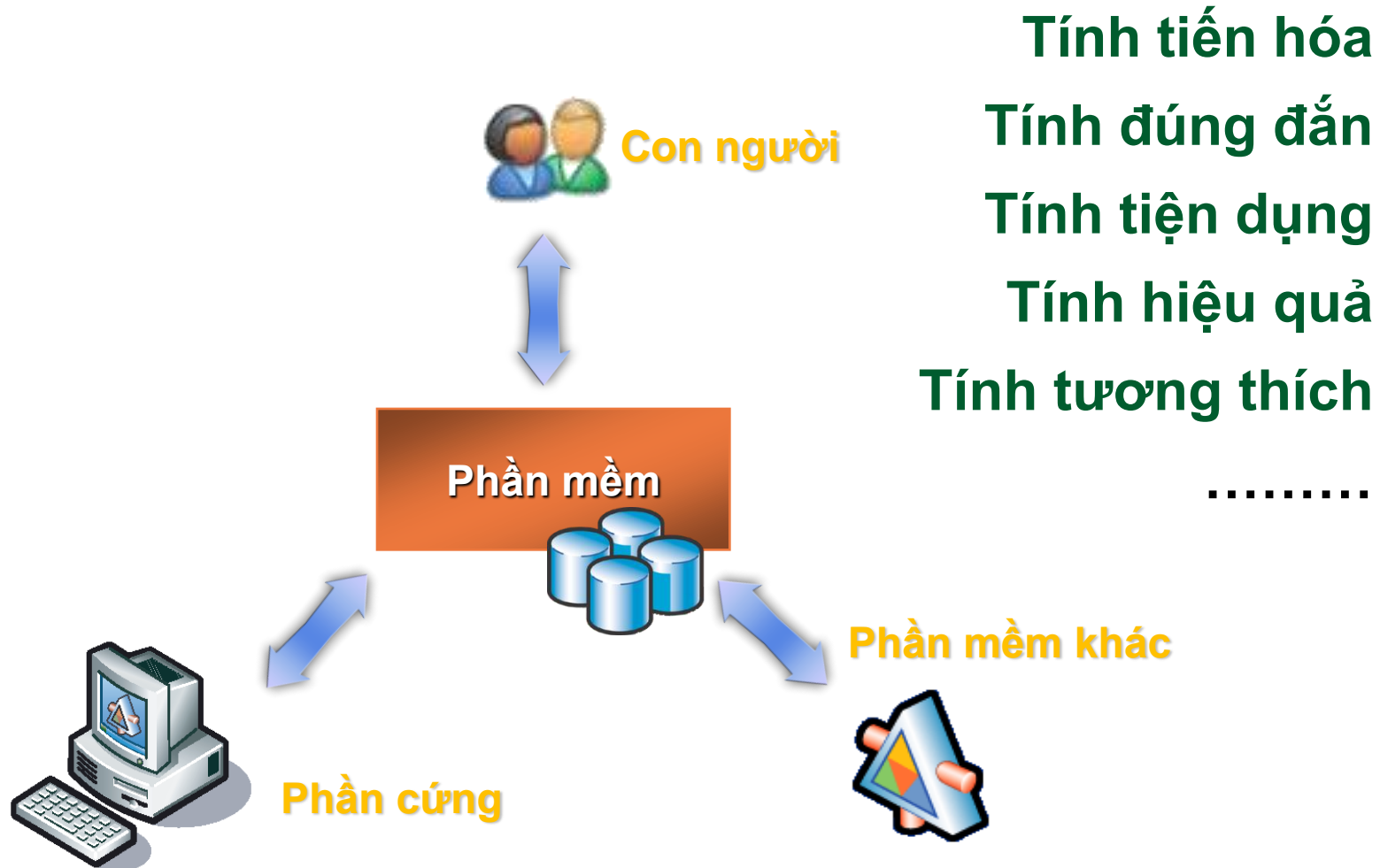
# Chất lượng phần mềm

Tính tiến hóa: một trong các tính chất quan trọng nhất được quan tâm xem xét trong ngành Công nghệ Phần mềm



Dưới góc nhìn của Người sử dụng

# Chất lượng phần mềm



Dưới góc nhìn của Người sử dụng



# Chất lượng phần mềm

## Tính dễ kiểm tra

việc kiểm tra các thành phần phù hợp với yêu cầu phần mềm là dễ dàng nhất có thể được

## Tính dễ sửa lỗi

khi có sự không phù hợp (so với yêu cầu) trong quá trình kiểm tra một thành phần, việc phát hiện chính xác “vị trí lỗi” và sửa lỗi là nhanh nhất có thể được

## Tính dễ bảo trì

khi cần nâng cấp, cải tiến một thành phần (theo yêu cầu mới), việc cập nhật phần mềm là nhanh, chính xác nhất có thể được và đặc biệt là cố gắng hạn chế ảnh hưởng đến các thành phần khác

Góc  
nhìn  
Chuyên  
viên  
tin  
học

# Chất lượng phần mềm

## Tính tái sử dụng

**các thành phần đã thực hiện có thể dùng lại trong các phần mềm cùng lớp (hoặc cùng lĩnh vực) với thời gian và công sức ít nhất có thể được**

Góc  
nhìn  
Chuyên  
viên  
tin  
học

# Cuộc khủng hoảng phần mềm

- Số lượng các phần mềm tăng vọt (do sự phát triển của phần cứng: tăng khả năng, giá thành hạ)
- Có quá nhiều **khuyết điểm** trong các phần mềm được dùng trong xã hội:
  - Thực hiện **không đúng yêu cầu** (tính toán sai, không ổn định...)
  - Thời gian **bảo trì** nâng cấp quá lâu, chi phí cao, hiệu quả thấp
  - Khó sử dụng, Thực hiện chậm
  - Không chuyển đổi dữ liệu giữa các phần mềm
  - .....

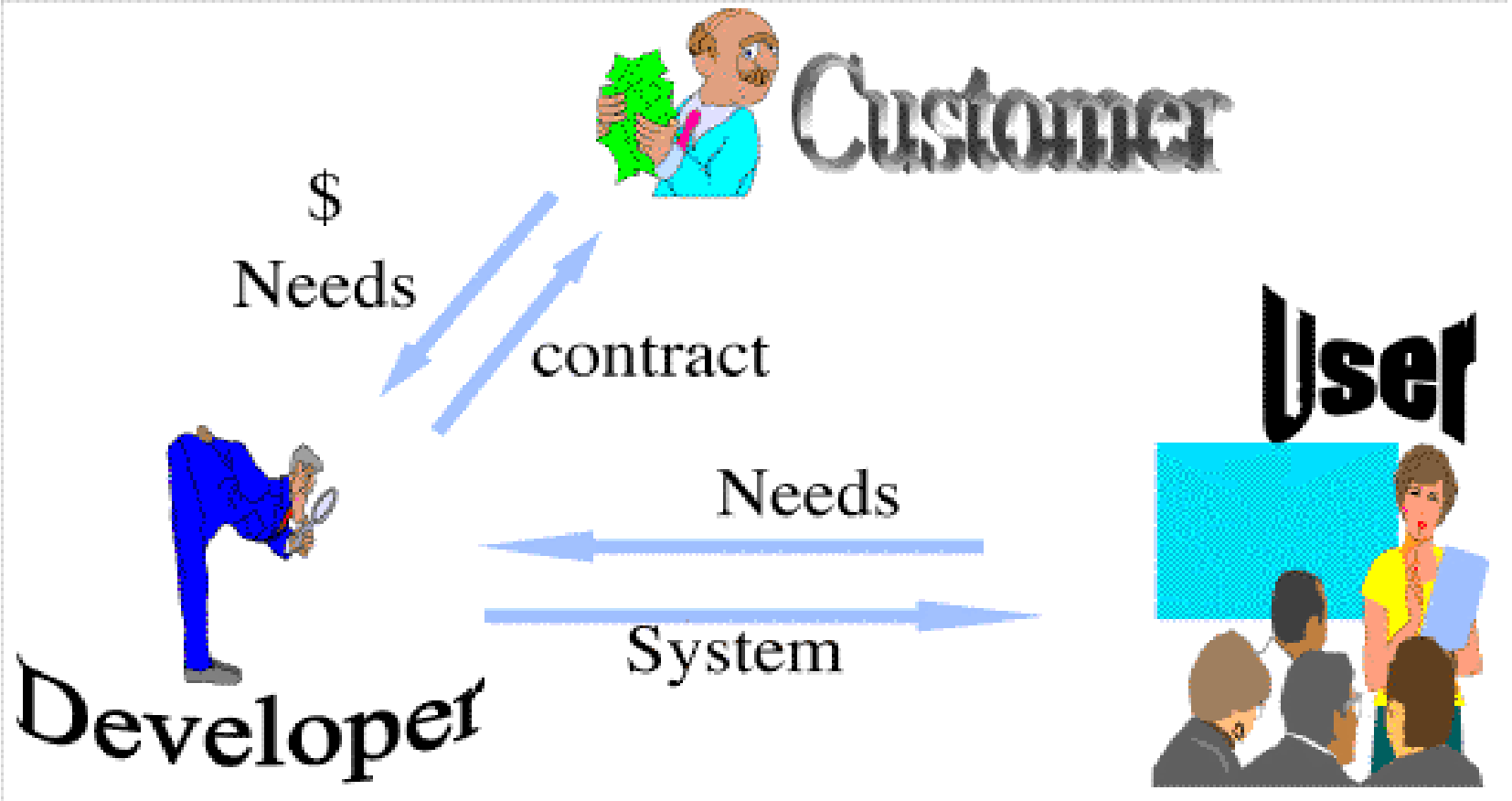
# Cuộc khủng hoảng phần mềm

## ■ Một số kết luận

- ❑ Việc tăng vọt số lượng phần mềm là điều hợp lý và sẽ còn tiếp diễn
- ❑ Các khuyết điểm của phần mềm có nguồn gốc chính từ **phương pháp, cách thức và quy trình** tiến hành xây dựng phần mềm
  - **Cảm tính**: mỗi người theo một phương pháp riêng
  - **Thô sơ, đơn giản**: chỉ tập trung vào việc lập trình mà ít quan tâm đến các công việc cần làm khác (khảo sát hiện trạng, phân tích yêu cầu, thiết kế...)
  - **Thủ công**: còn thiếu các công cụ hỗ trợ quy trình phát triển

# Thảo luận

Customer-driven



Market-driven

# In thinking about the process of developing software

- Who is involved? What roles do individuals have?
- How many individuals are involved?
- What are the tasks?
- How are tasks allocated to people?
- What are the characteristics of those engaged in a given task?
- Is there a match between the characteristics of the individual and the demands of the task?
- How is the process managed, coordinated, controlled?
- What are the work products produced during development?
- How long do the various tasks take?
- How do we know a task is complete?
- Are there tasks that can be performed concurrently?
- What communications support is required to facilitate people working together?

# In thinking about the product resulting from the development activity

- What is the functionality provided?
- What are the intended use(s) of the product in the customer's environment?
- What expectations does the customer have for the product?
- Is the product operating reasonably in the use environment?
- What considerations are there for system reliability?
- How will the product be maintained over time?
- What performance characteristics do the various functions need?
- How does the system interface with the users?
- Are the resources involved with system use being used wisely?
- How many defects are present in the product?
- What quality attributes have been assigned to this product and how should they be quantified?