

# CHƯƠNG 1: LƯỢC SỬ VÀ MỤC TIÊU CỦA CÔNG NGHỆ PHẦN MỀM

# I. Mục tiêu

- **Kiến thức:** Giúp sinh viên
- **Kỹ năng:**
  - Kỹ năng tìm tòi, nghiên cứu tài liệu.
- **Thái độ:**
  - Nghiêm túc trong công việc, ham học hỏi, tự học để nâng cao trình độ.

## II. Chuẩn bị

- Giáo viên:
  - Giáo án, máy tính, projector.
- Sinh viên:
  - Tài liệu, vở, viết.

## III. Phương pháp

- Phương pháp thuyết trình
- Phương pháp đặt vấn đề.

# Chương 1. LƯỢC SỬ VÀ MỤC TIÊU CỦA CÔNG NGHỆ PHẦN MỀM

- 1.1. Kỹ thuật lập trình giai đoạn thế hệ thứ nhất của máy tính điện tử.
- 1.2. Cuộc khủng hoảng lập trình những năm 60.
- 1.3. Công nghệ phần mềm là gì?
- 1.4. Những nội dung cơ bản của Công nghệ phần mềm.

## 1.1. Kỹ thuật lập trình giai đoạn thế hệ thứ nhất của máy tính điện tử

- Năng suất làm các sản phẩm phần mềm là thấp nhất so với các hoạt động trí tuệ khác.
- Việc kiểm tra, tìm lỗi và hoàn thiện sản phẩm mất rất nhiều thời gian.
- Trước những năm 60 người ta coi lập trình như là hoạt động nghệ thuật nhuộm màu sắc tài nghệ cá nhân hơn là khoa học.

## 1.2. Cuộc khủng hoảng lập trình những năm 60

- Chi phí cho tin học quá lớn, trong đó chi phí cho phần mềm chiếm tỷ lệ cao và ngày càng tăng so với chi phí kỹ thuật tin học (phần cứng).
- Rất nhiều đề án lớn nhằm ứng dụng tin học bị thất bại liên tiếp. Nguyên nhân thất bại chủ yếu là do phần đảm bảo chương trình.

## 1.3. Công nghệ phần mềm là gì?

- Khi tìm hiểu về công nghệ phần mềm, chúng ta thường đặt ra một số câu hỏi sau:
  - Phần mềm là gì?
  - Công nghệ phần mềm là gì?
  - Sự khác biệt giữa công nghệ phần mềm và khoa học máy tính?
  - Sự khác biệt giữa công nghệ phần mềm và công nghệ hệ thống?
  - Quy trình phần mềm là gì?
  - Mô hình quy trình phát triển phần mềm là gì?
  - Chi phí của công nghệ phần mềm bao gồm những gì?
  - Các phương pháp công nghệ phần mềm là gì?
  - CASE (Computer-Aided Software Engineering) là gì?
  - Thế nào là một phần mềm tốt?

## 1.3. Công nghệ phần mềm là gì? (2)

1.3.1. Định nghĩa

1.3.2. Triển khai phần mềm theo sắc thái công nghệ.

1.3.3. Đảm bảo sự thông hiểu lẫn nhau.

1.3.4. Tận dụng quỹ thuật toán và chương trình.

1.3.5. Phần mềm cho công nghệ phần mềm.

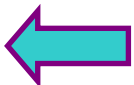


## 1.3.1. Định nghĩa

- Khái niệm về công nghệ phần mềm được đưa ra lần đầu tiên vào năm 1968 tại hội nghị thảo luận về khủng hoảng phần mềm.
- Công nghệ phần mềm đề cập tới các lý thuyết, phương thức và công cụ để xây dựng phần mềm chuyên nghiệp, mang lại lợi nhuận cao.

## 1.3.1. Định nghĩa (2)

- Công nghệ phần mềm (Software Engineering) là lĩnh vực nghiên cứu của Tin học nhằm đề xuất các nguyên lý, phương pháp, công cụ, cách tiếp cận và phương tiện phục vụ cho việc thiết kế và cài đặt các sản phẩm phần mềm nhằm đạt được các chỉ tiêu cơ bản sau đây một cách tốt nhất:
  - Tính đúng đắn,
  - Tính khoa học,
  - Tính dễ học và dễ sửa đổi, cải tiến,
  - Tính dễ sử dụng,
  - Tính phổ dụng,
  - Tính độc lập với thiết bị.



## 1.3.2. Triển khai PM theo sắc thái công nghệ

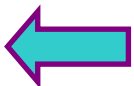
- Công nghệ sản xuất một mặt hàng nào đó thường gồm các giai đoạn sau:
  1. Tìm hiểu nhu cầu của khách hàng.
  2. Xác định các chức năng cần có của sản phẩm.
  3. Chia các chức năng thành những nhóm độc lập tương đối với nhau. Mỗi nhóm ứng với một bộ phận hợp thành sản phẩm.
  4. Giao việc thiết kế và sản xuất từng bộ phận cho từng người hoặc nhóm người.
  5. Các nhóm triển khai công việc: Thực hiện các bước thiết kế, sản xuất, thử nghiệm.

### 1.3.2. Triển khai PM theo sắc thái công nghệ (2)

- Công nghệ sản xuất một mặt hàng nào đó thường gồm các giai đoạn sau:
  6. Ghép nối các bộ phận, chi tiết thành sản phẩm.
  7. Thử nghiệm sản phẩm, sửa nếu cần.
  8. Bán hoặc giao lô sản phẩm đầu tiên cho khách hàng.
  9. Thu thập thông tin phản hồi từ phía người sử dụng. Quyết định sửa lại sản phẩm, cải tiến hoặc hủy bỏ việc sản xuất sản phẩm này.

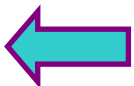
01/15/10

Bài giảng môn CNPM

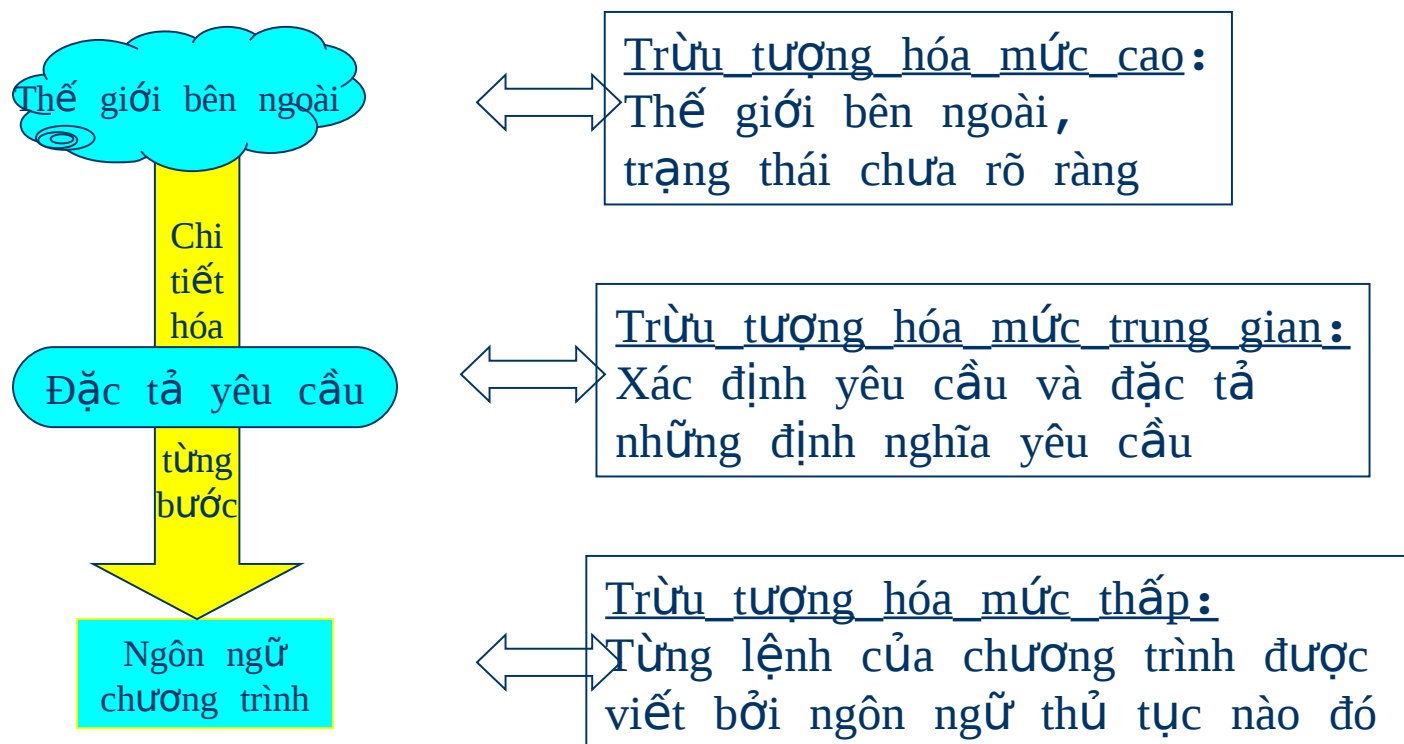


### 1.3.3. Đảm bảo sự thông hiểu lẫn nhau

- Là một trong những yếu tố quan trọng hàng đầu.
- Phương tiện giao tiếp trong nhóm chính là ngôn ngữ đặc tả.
- Các nguyên tắc hành động đảm bảo trong sáng, dễ hiểu: triển khai chương trình theo cấu trúc, từ trên xuống và tinh chế dần.



## Cách tiếp cận từ trên xuống (top-down approach)



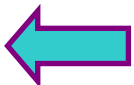
### 1.3.4. Tận dụng quỹ thuật toán và chương trình

- Tìm và sử dụng lại những module, thủ tục hoặc thuật toán đã có sẵn.
- Hãy làm việc theo nguyên tắc “mình vì mọi người, mọi người vì mình”.



### 1.3.5. Phần mềm cho công nghệ phần mềm

- CASE (Computer-Aided Software Engineering) là những hệ chương trình được sử dụng để hỗ trợ các hoạt động trong quy trình xây dựng phần mềm.
- CASE trợ giúp các nhóm làm phần mềm trong tất cả các giai đoạn. Có 2 loại CASE:
  - **Upper-CASE**: công cụ để hỗ trợ các hoạt động đầu tiên như đặc tả yêu cầu và thiết kế.
  - **Lower-CASE**: công cụ để hỗ trợ các hoạt động sau như lập trình, gỡ lỗi và kiểm thử.





## 1.4. Những nội dung cơ bản của CNPM

1.4.1. Tìm hiểu và phân tích yêu cầu

1.4.2. Đặc tả chương trình

1.4.3. Lập trình cấu trúc

1.4.4. Kiểm thử

1.4.5. Chứng minh tính đúng đắn của chương trình.

1.4.6. Xem xét chất lượng chương trình.

1.4.7. Quản lý thiết kế, cài đặt, lập hồ sơ.

1.4.8. Đảm bảo việc cung cấp các phần mềm trợ giúp cho hoạt động của nhóm thiết kế.

# MỘT CÁCH NHÌN TỔNG QUAN VỀ QUY TRÌNH PHÁT TRIỂN PHẦN MỀM

Gồm 3 giai đoạn lớn:

- Giai đoạn định nghĩa: Phân tích hệ thống, Hoạch định đề tài, Phân tích yêu cầu.
- Giai đoạn phát triển: Thiết kế phần mềm, Sinh mã, Kiểm tra phần mềm.
- Giai đoạn bảo trì.

## Ví dụ: Trình tự giải quyết vấn đề từ mức thiết kế chương trình đến mức lập trình

- Bài toán: từ một nhóm  $N$  số khác nhau tăng dần, hãy tìm số có giá trị bằng  $K$  (nhập từ ngoài vào) và in ra vị trí của nó?
- Giải từng bước từ khái niệm đến chi tiết hóa từng câu lệnh bởi ngôn ngữ lập trình nào đó.
- Chọn giải thuật tìm kiếm nhị phân (pp nhị phân)

## Cụ thể hóa thủ tục qua các chức năng

Bài toán đã cho

Nhập giá trị K

Nhập giá trị nhóm N số

Tìm kiếm giá trị (pp nhị phân)

In ra vị trí (nếu có)

## Cụ thể hóa bước tiếp theo

Tìm kiếm giá trị  
(pp nhị phân)

Xác lập phạm vi mảng số

Lặp lại xử lý tìm kiếm giá trị K trong phạm vi tìm kiếm

Lặp lại tìm kiếm K trong phạm vi tìm kiếm

Tìm vị trí giữa phân đôi mảng

So sánh K với giá trị giữa

Đặt lại phạm vi tìm kiếm

## Mức mô tả chương trình:

Bắt đầu

Đọc K

Nhận giá trị cho mảng 1 chiều A(I), (I = 1, 2, . . . , N)

MIN = 1

MAX = N

DO WHILE (Có giá trị bằng K không, cho đến khi MIN > MAX)

Lấy  $MID = (MIN + MAX) / 2$

IF A(MID) > K THEN

MAX = MID - 1

ELSE

IF A(MID) < K THEN

MIN = MID + 1

ELSE

In giá trị MID

ENDIF

ENDIF

ENDDO

KếtThúc

# BÀI TẬP CHƯƠNG 1

## 1. Sự khác biệt giữa công nghệ phần mềm và khoa học máy tính?

Khoa học máy tính đề cập tới lý thuyết và những vấn đề cơ bản;

Còn công nghệ phần mềm đề cập tới các nguyên lý, phương thức và công cụ để xây dựng phần mềm chuyên nghiệp, hữu ích.

# BÀI TẬP CHƯƠNG 1

## 2. Sự khác biệt giữa công nghệ phần mềm và công nghệ hệ thống?

Công nghệ hệ thống (hay còn gọi là kỹ nghệ hệ thống) liên quan tới tất cả các khía cạnh của quá trình phát triển hệ thống dựa máy tính bao gồm: phần cứng, phần mềm, và công nghệ xử lý;

Còn Công nghệ phần mềm chỉ là một phần của quy trình này, nó có liên quan tới việc phát triển hạ tầng phần mềm, điều khiển, các ứng dụng và cơ sở dữ liệu trong hệ thống.



# BÀI TẬP CHƯƠNG 1

## 3. Quy trình phần mềm là gì ?

Quy trình phần mềm là một tập hợp các hành động mà mục đích của nó là xây dựng và phát triển phần mềm. Những hành động thường được thực hiện trong các quy trình phần mềm bao gồm:

- Đặc tả.
  - Phát triển.
  - Kiểm thử.
  - Mở rộng.
- Những loại hệ thống khác nhau sẽ cần những quy trình phát triển khác nhau.

# BÀI TẬP CHƯƠNG 1

## 4. Mô hình quy trình phát triển phần mềm là gì?

- ✚ Là một thể hiện đơn giản của một quy trình phần mềm, nó được biểu diễn từ một góc độ cụ thể.
- ✚ Ví dụ:
  - Mô hình luồng dữ liệu (data-flow)
  - Mô hình Vai trò/Hành động (Role/action)
  - Mô hình thác nước (waterfall)
  - Mô hình phát triển lặp lại (Iterative development)
  - Mô hình công nghệ phần mềm dựa thành phần (Component-based software engineering).

# BÀI TẬP CHƯƠNG 1

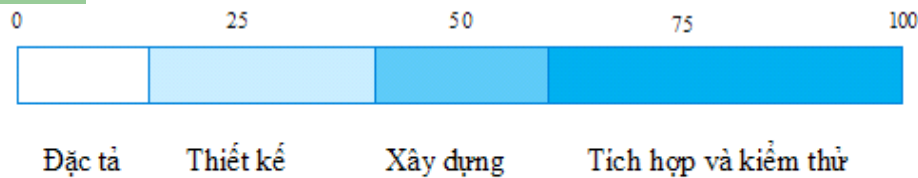
5. Xây dựng một sản phẩm phần mềm phải đầu tư những chi phí nào ?



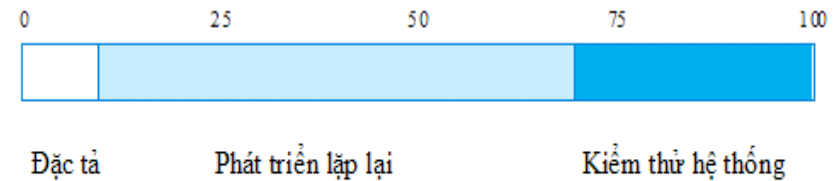
*Hình 1.1: Chi phí sản xuất phần mềm*

# BÀI TẬP CHƯƠNG 1

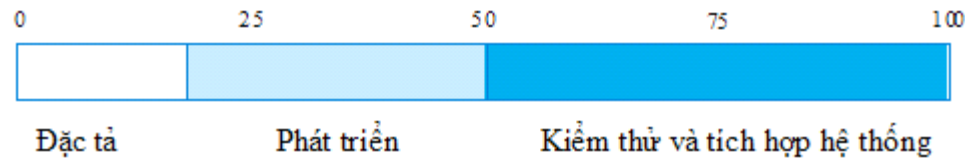
## 5. Xây dựng một sản phẩm phần mềm phải đầu tư những chi phí nào ?



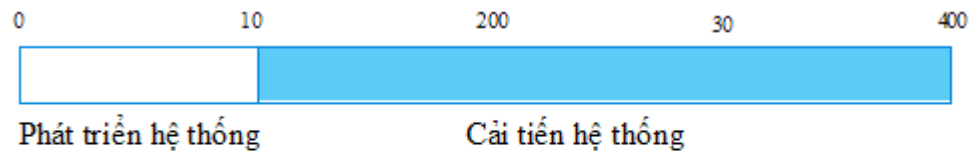
Hình 1.2: Phân bổ chi phí trong mô hình thác nước



Hình 1.3: Phân bổ chi phí trong mô hình phát triển lặp lại



Hình 1.4: Phân bổ chi phí trong công nghệ phần mềm hướng thành phần



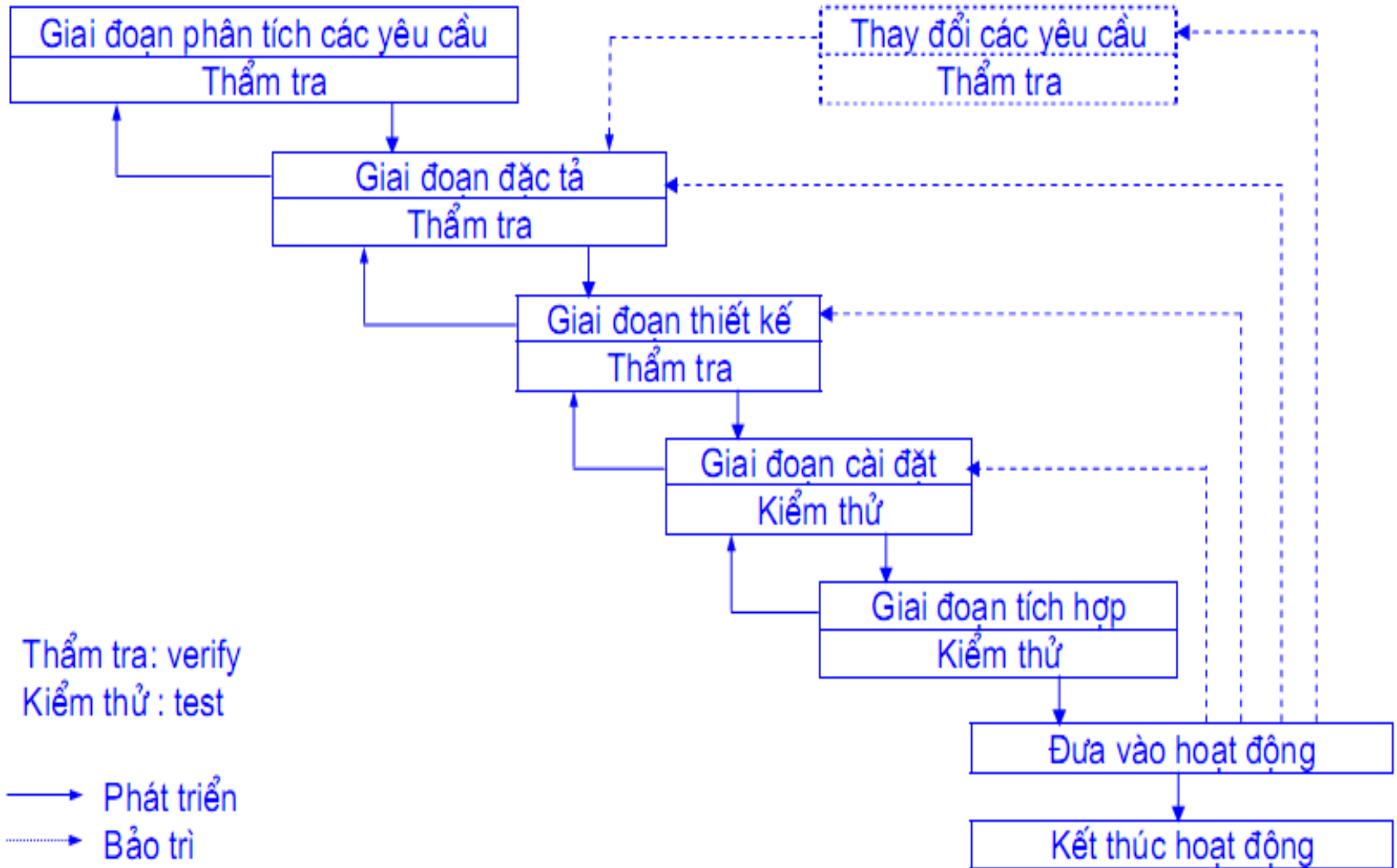
Hình 1.5: Phân bổ chi phí trên các hệ thống có chu kỳ sống dài

# BÀI TẬP CHƯƠNG 1

## 6. Các mô hình phát triển phần mềm:

- ✚ Năm mô hình phát triển phần mềm phổ biến thường được sử dụng:
  - Mô hình thác nước
  - Mô hình xoắn ốc
  - Mô hình xây dựng tiến triển
  - Công nghệ phần mềm dựa thành phần
  - Mô hình phát triển lặp lại, tăng thêm

# MÔ HÌNH THÁC NƯỚC



## MÔ HÌNH THÁC NƯỚC (2)

- Mô hình thác nước do Royce đề xuất
- Đặc điểm: Các pha trên phải được thực hiện một cách tuần tự; kết thúc pha trước, rồi mới được thực hiện pha tiếp theo.

## MÔ HÌNH THÁC NƯỚC (2)

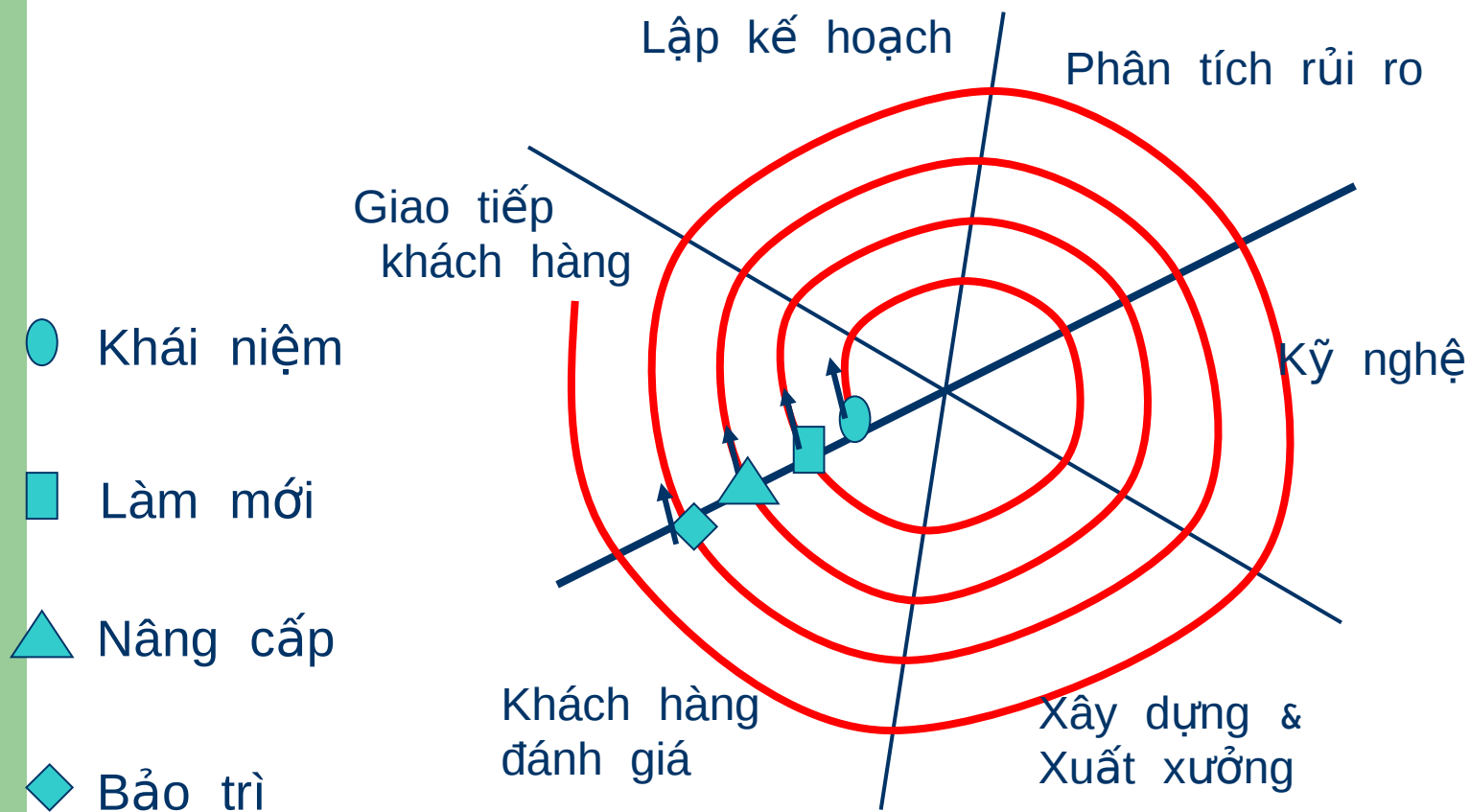
- Ưu điểm:
  - Kỷ luật cao, quy định tốt về tài liệu cho mỗi giai đoạn, được kiểm chứng cẩn thận.
- Nhược điểm:
  - Rất khó khăn trong việc thay đổi các pha đã được thực hiện.
  - Quá nhiều kiểm thử, thẩm tra và tài liệu.
  - Tài liệu khó hình dung và khó hiểu với khách hàng
  - Mô hình này chỉ thích hợp khi các yêu cầu đã được tìm hiểu rõ ràng và những thay đổi sẽ được giới hạn một cách rõ ràng trong suốt quá trình thiết kế.



# MÔ HÌNH XOẮN ỐC

- Quy trình phát triển phần mềm được biểu diễn như một vòng xoắn ốc. Các pha trong quy trình phát triển xoắn ốc bao gồm:
  - Giao tiếp khách hàng: giữa người phát triển và khách hàng để tìm hiểu yêu cầu, ý kiến.
  - Lập kế hoạch: Xác lập tài nguyên, thời hạn và thông tin khác.
  - Phân tích rủi ro: Xem xét mạo hiểm kỹ thuật, mạo hiểm quản lý và thực hiện các hành động để giảm thiểu rủi ro.
  - Kỹ nghệ: Xây dựng một hay một số biểu diễn của ứng dụng.
  - Xây dựng và xuất xưởng: xây dựng, kiểm thử, cài đặt và cung cấp hỗ trợ người dùng (tư liệu, huấn luyện, . . .).
  - Đánh giá của khách hàng: Nhận các phản hồi của người sử dụng về biểu diễn phần mềm trong giai đoạn kỹ nghệ và cài đặt.

# MÔ HÌNH XOẮN ỐC (2)



## MÔ HÌNH XOẮN ỐC (3)

- Điểm mạnh:
  - Dễ kiểm soát các mạo hiểm ở từng mức tiến hóa. Đánh giá mức độ rủi ro.
  - Mục tiêu quan trọng luôn là chất lượng phần mềm.
  - Giảm nhẹ kiểm thử và nhanh chóng sửa chữa những lỗi xảy ra.
  - Bảo trì đơn giản: chỉ là một vòng tròn trong xoắn ốc, không có sự phân biệt trong phát triển và bảo trì.

## MÔ HÌNH XOẪN ỐC (4)

- Điểm yếu:
  - Dành riêng cho các phần mềm nội bộ có kích thước lớn.
  - Vì dự án có thể bị chấm dứt do các đánh giá về rủi ro, dẫn đến một số hậu quả...
  - Các nhà phát triển phải có khả năng phân tích rủi ro và giải quyết rủi ro.

# MÔ HÌNH XÂY DỰNG TIẾN TRIỂN

- Mô hình xây dựng tiến triển dựa trên ý tưởng xây dựng một **mẫu thử ban đầu** và đưa cho người sử dụng xem xét;
- Sau đó, **tinh chỉnh** mẫu thử qua nhiều phiên bản cho đến khi thoả mãn yêu cầu của người sử dụng thì dừng lại.

## MÔ HÌNH XÂY DỰNG TIẾN TRIỂN (2)

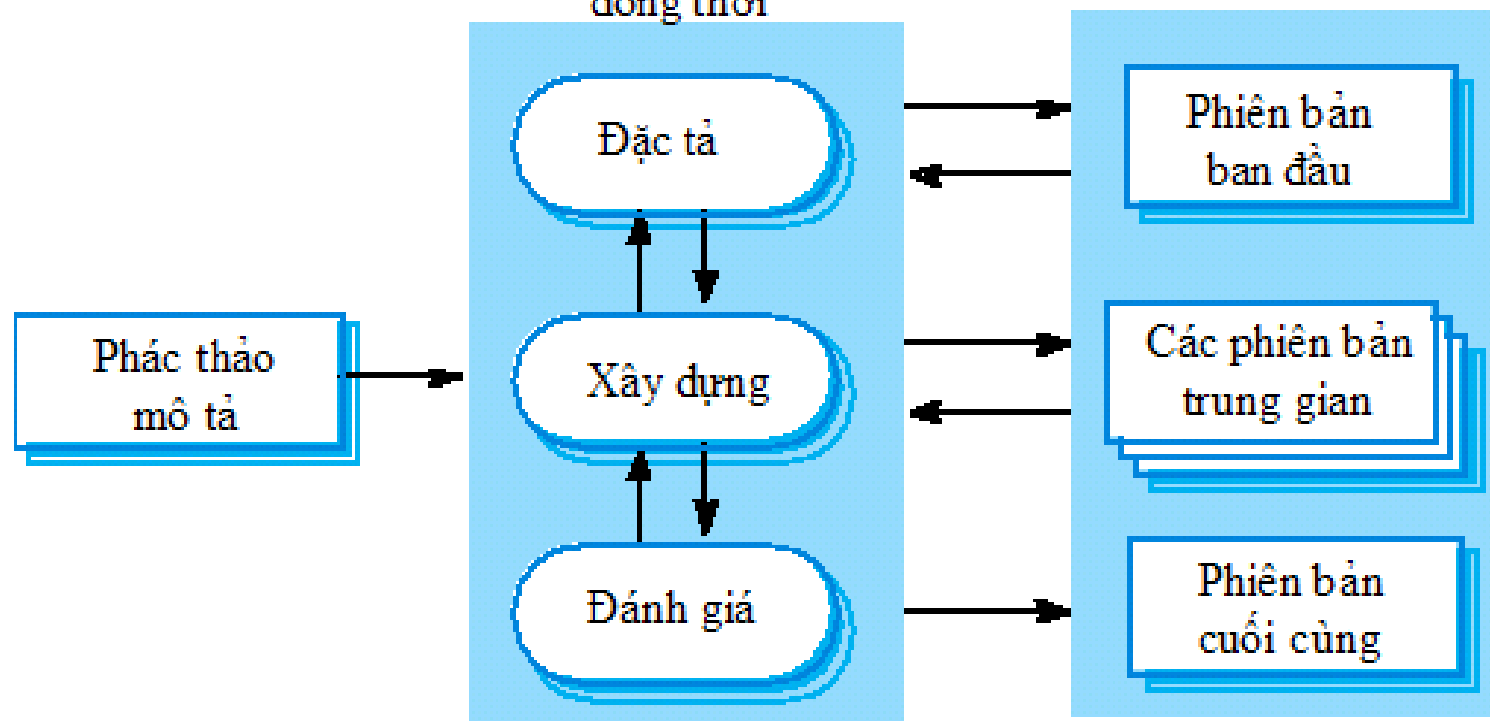
- Có hai phương pháp để thực hiện mô hình này:
  - **Phát triển thăm dò**: mục đích là để làm việc với khách hàng và để đưa ra hệ thống cuối cùng từ những đặc tả sơ bộ ban đầu.
    - + Phương pháp này thường bắt đầu thực hiện với những **yêu cầu được tìm hiểu rõ ràng** và sau đó, bổ sung những đặc điểm mới được đề xuất bởi khách hàng.
    - + Cuối cùng, khi các yêu cầu của người sử dụng được thoả mãn thì cũng là lúc chúng ta đã xây dựng xong hệ thống.

## MÔ HÌNH XÂY DỰNG TIẾN TRIỂN (3)

- Có hai phương pháp để thực hiện mô hình này:
  - **Loại bỏ mẫu thử**: mục đích là để tìm hiểu các yêu cầu của hệ thống.
    - + Phương pháp này thường bắt đầu với những **yêu cầu không rõ ràng và ít thông tin**. Các mẫu thử sẽ được xây dựng và chuyển giao tới cho người sử dụng.
    - + Từ đó, ta có thể phân loại những yêu cầu nào là thực sự cần thiết và lúc này mẫu thử không còn cần thiết nữa.
    - + Như vậy, mẫu thử chỉ có tác dụng để làm sáng tỏ yêu cầu của người sử dụng.

# MÔ HÌNH YÊU DÙNG TIẾN TIẾN (1)

Các hoạt động  
đồng thời





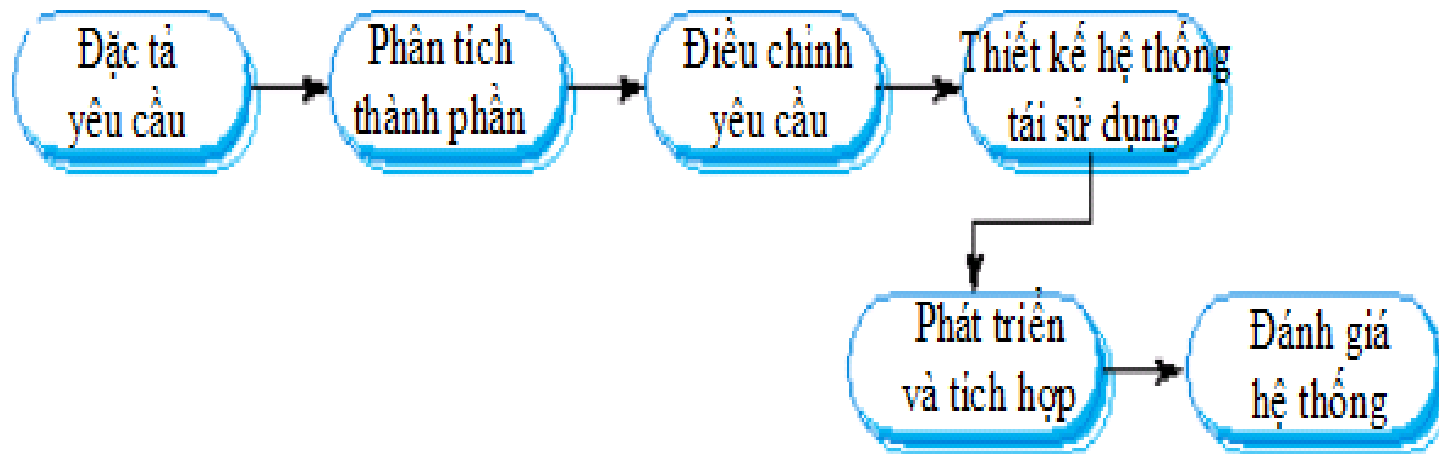
## MÔ HÌNH XÂY DỰNG TIẾN TRIỂN (5)

- Nhược điểm:
  - Thiếu tầm nhìn của cả quy trình;
  - Các hệ thống thường hướng cấu trúc nghèo nàn;
  - Yêu cầu các kỹ năng đặc biệt (Ví dụ: các ngôn ngữ để tạo ra mẫu thử nhanh chóng).
  - Chỉ nên áp dụng với những hệ thống có tương tác ở mức độ nhỏ hoặc vừa; trên một phần của những hệ thống lớn; hoặc những hệ thống có thời gian chu kỳ tồn tại ngắn.

# CÔNG NGHỆ PHẦN MỀM DỰA THÀNH PHẦN

- Mô hình này dựa trên kỹ thuật tái sử dụng một cách có hệ thống; trong đó hệ thống được tích hợp từ nhiều thành phần đang tồn tại hoặc các thành phần thương mại COTS (Commercial-off-the-shelf).
- Các trạng thái chính của quy trình bao gồm:
  - Phân tích thành phần sẵn có
  - Điều chỉnh yêu cầu
  - Thiết kế hệ thống với kỹ thuật tái sử dụng
  - Xây dựng và tích hợp hệ thống

# CÔNG NGHỆ PHẦN MỀM DỰA THÀNH PHẦN (2)

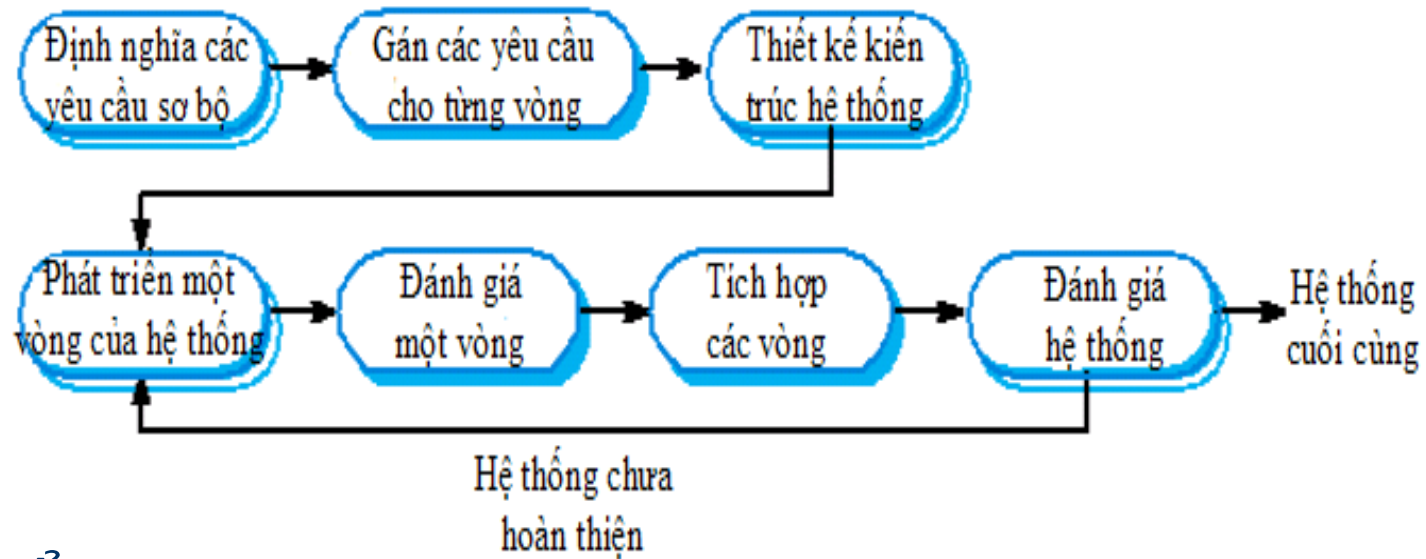


- Gắn với những công nghệ hướng đối tượng (Object-oriented technologies) qua việc tạo các lớp có chứa cả dữ liệu và giải thuật xử lý dữ liệu.
- Có nhiều tương đồng với mô hình xoắn ốc.
- Với ưu điểm tái sử dụng các thành phần qua Thư viện/kho các lớp: tiết kiệm 70% thời gian, 80% giá thành, chỉ số sản xuất 26.2/16.9.

## MÔ HÌNH PHÁT TRIỂN LẶP LẠI, TĂNG THÊM

- Mô hình này được đề xuất dựa trên ý tưởng thay vì phải xây dựng và chuyển giao hệ thống một lần thì sẽ được **chia thành nhiều vòng, tăng dần**. Mỗi vòng là một phần kết quả của một chức năng được yêu cầu.
- Các yêu cầu của người sử dụng được đánh **thứ tự ưu tiên**. Yêu cầu nào có thứ tự ưu tiên càng cao thì càng ở trong những vòng phát triển sớm hơn.

## MÔ HÌNH PT LẶP LẠI, TĂNG THÊM (2)



### Ưu điểm:

- Sau mỗi lần tăng vòng thì có thể chuyển giao kết quả thực hiện được cho khách hàng nên các chức năng của hệ thống có thể nhìn thấy sớm hơn.
- Các vòng trước đóng vai trò là mẫu thử để giúp tìm hiểu thêm các yêu cầu ở những vòng tiếp theo.
- Những chức năng của hệ thống có thứ tự ưu tiên càng cao thì sẽ được kiểm thử càng kỹ