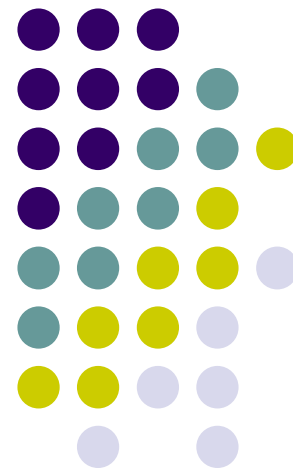


# NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

---

## PHẦN II – TIỀN TRÌNH PHẦN MỀM

Bộ môn Công nghệ phần mềm,  
Khoa CNTT&TT, Đại học Cần Thơ



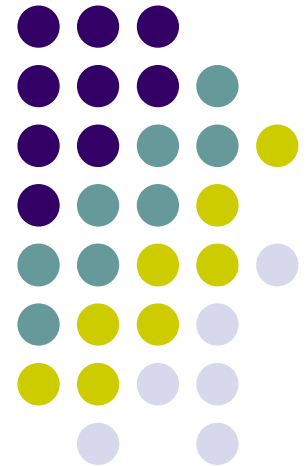


# Nội dung

- Phân tích và Đặc tả
- Thiết kế
- Lập trình
- Kiểm thử
- Triển khai hệ thống và Bảo trì

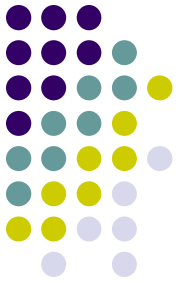
# TIẾN TRÌNH PHẦN MỀM

## PHẦN II.4 – KIỂM THỬ



# Nội dung

- Phần 1 – Kiểm thử chương trình
- Phần 2 – Kiểm thử hệ thống





# Phần 1 - Nội dung

- Các lỗi phần mềm
- Các vấn đề trong kiểm thử
- Kiểm thử đơn vị
- Kiểm thử tích hợp
- Lập kế hoạch kiểm thử
- Các công cụ kiểm thử tự động

# Lỗi phần mềm



- Các nguyên nhân làm phần mềm thất bại
  - Đặc tả sai
  - Đặc tả thiếu
  - Yêu cầu không thể thực thi
  - Thiết kế không đúng hoặc được cài đặt không đúng
  - Mã lệnh có thiếu sót

# Lỗi phần mềm



- Mục đích của kiểm thử: phát hiện ra các lỗi.
- Một kiểm thử là thành công chỉ khi lỗi được phát hiện.
  - Nhận dạng lỗi là quá trình xác định lỗi nào đã gây ra sự thất bại.
  - Hiệu chỉnh lỗi là quá trình tạo ra các thay đổi đối với hệ thống nhằm loại bỏ các lỗi.

# Lỗi phần mềm



- Các loại lỗi
  - Lỗi giải thuật (các lỗi giải thuật điển hình)
    - Lỗi giải thuật xuất hiện khi logic hay giải thuật của thành phần không tạo ra kết xuất đúng với đầu vào xác định vì đã có sai sót trong các bước xử lý như:
      - Rẽ nhánh quá sớm
      - Rẽ nhánh quá trễ
      - Kiểm thử cho điều kiện sai
      - Quên khởi tạo giá trị ban đầu của biến hay đặt vòng lặp bất biến
      - Quên kiểm thử cho điều kiện đặc biệt
      - So sánh giá trị của các kiểu dữ liệu không phù hợp
    - Lỗi cú pháp



# Lỗi phần mềm



- Các loại lỗi
  - Lỗi về độ chính xác và tính toán
    - Sự thực hiện của một công thức là sai hoặc không tính ra kết quả với độ chính xác mong muốn.
  - Lỗi tài liệu
    - Tài liệu không phù hợp với cái mà chương trình làm.
  - Lỗi về quá tải hay dung lượng
    - Sự thực hiện của hệ thống là không thể chấp nhận khi đạt đến các ranh giới nào đó.

# Lỗi phần mềm



- Các loại lỗi
  - Lỗi hợp tác và thời gian.
  - Lỗi thực hiện.
    - Hệ thống không thực hiện với tốc độ được mô tả
  - Lỗi quy trình và chuẩn.

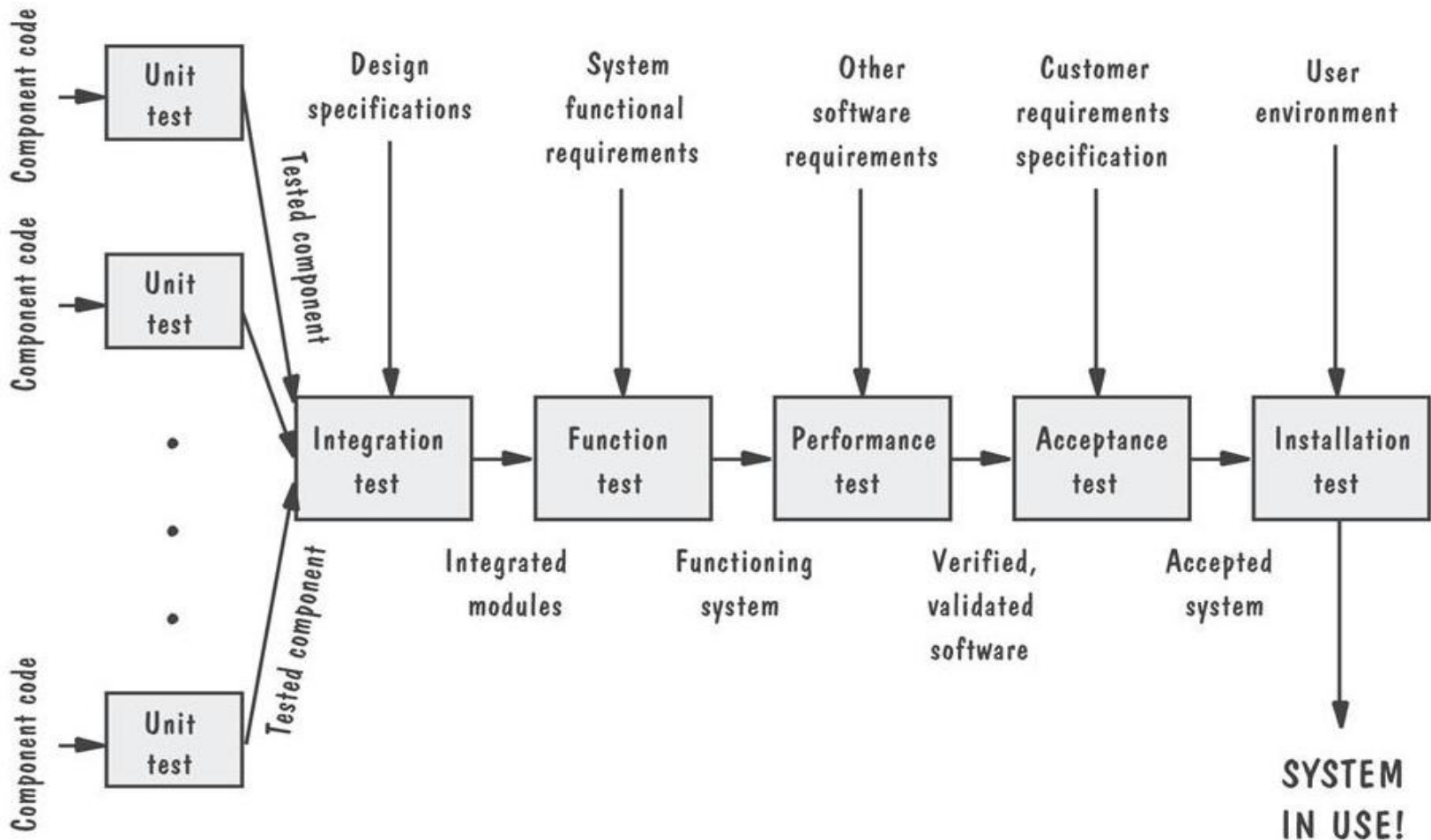


# Các vấn đề trong kiểm thử

- Tổ chức kiểm thử
  - Kiểm thử đơn vị
  - Kiểm thử tích hợp
  - Kiểm thử chức năng
  - Kiểm thử sự thực hiện
  - Kiểm thử chấp nhận
  - Kiểm thử cài đặt

# Các vấn đề trong kiểm thử

- Tổ chức kiểm thử



# Các vấn đề trong kiểm thử



- Quan điểm kiểm thử
  - Các chương trình được xem như những thành phần của một hệ thống lớn hơn, không như tài sản của những người đã viết ra chúng.



# Các vấn đề trong kiểm thử

- Người kiểm thử: nhóm kiểm thử độc lập
  - Tránh sự mâu thuẫn.
  - Tăng tính khách quan.
  - Cho phép kiểm thử và lập trình đồng thời.

# Các vấn đề trong kiểm thử

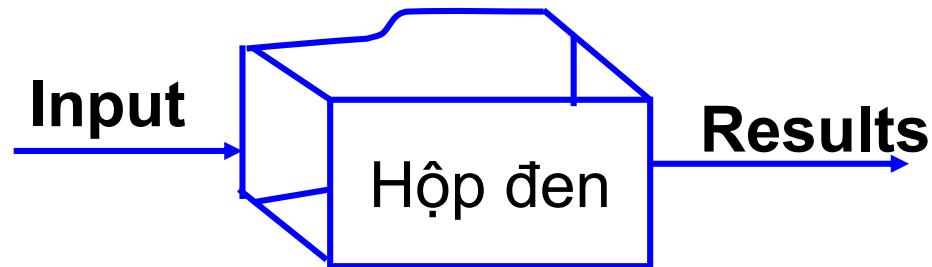


- Tổng quan về các phương pháp kiểm thử
  - Kiểm thử hộp đóng hay hộp đen: sử dụng **chức năng** của đối tượng kiểm thử để kiểm thử theo nhiều cách.
  - Kiểm thử hộp mở hay hộp trắng: sử dụng **cấu trúc** của đối tượng kiểm thử để kiểm thử theo nhiều cách.

# Các vấn đề trong kiểm thử



- Kiểm thử hộp đen
  - Đặc điểm
    - Đối tượng kiểm thử như một hộp đen, thông qua giao diện để đưa dữ liệu vào và nhận dữ liệu ra.



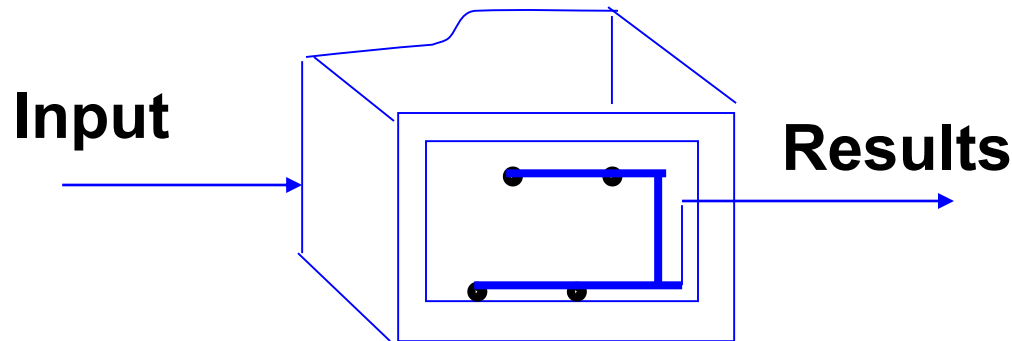
- Độc lập với các ràng buộc bị tác động bởi cấu trúc bên trong và tính logic của đối tượng.



# Các vấn đề trong kiểm thử



- Kiểm thử hộp trắng
  - Phương pháp thiết kế trường hợp kiểm thử sử dụng cấu trúc điều khiển của thiết kế thuật toán để dẫn ra các trường hợp kiểm thử.





# Các vấn đề trong kiểm thử

- Các yếu tố ảnh hưởng đến việc chọn phương pháp kiểm thử:
  - Số đường logic có thể có.
  - Trạng thái của dữ liệu input.
  - Số lượng tính toán có liên quan.
  - Độ phức tạp của giải thuật.

# Kiểm thử đơn vị



- Kiểm thử đơn vị: kiểm thử từng thành phần hay từng mô đun của phần mềm.
- Các bước kiểm thử đơn vị
  - Xem lại mã lệnh.
  - Biên dịch mã lệnh và loại bỏ bất cứ lỗi cú pháp nào còn lại
  - Phát triển các trường hợp kiểm thử để chỉ ra rằng input được chuyển đổi chính xác thành output mong đợi.

# Kiểm thử đơn vị



- Các bước chọn trường hợp kiểm thử
  - Xác định các mục tiêu kiểm thử
  - Lựa chọn các trường hợp kiểm thử
  - Định nghĩa một kiểm thử

# Kiểm thử đơn vị



- Tính toàn diện của kiểm thử
  - Kiểm thử câu lệnh
  - Kiểm thử rẽ nhánh
  - Kiểm thử đường đi
  - Kiểm thử sử dụng sự định nghĩa
  - Kiểm thử tất cả sử dụng
  - Kiểm thử tất cả sử dụng vị từ / kiểm thử một số sử dụng tính toán
  - Kiểm thử một số sử dụng vị từ / kiểm thử tất cả sử dụng tính toán

# Kiểm thử đơn vị - Kiểm thử đường đi



- Kiểm thử đường đi: Kiểm thử tất cả các hướng đi có thể
  - Thiết lập đồ thị dòng chảy.
  - Liệt kê các đường thực thi độc lập cơ bản.
  - Sinh các trường hợp kiểm thử cho các đường thực thi đó.

# Kiểm thử đường đi

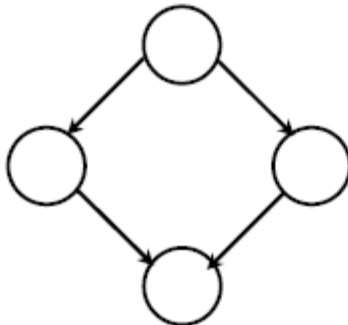


- Thiết lập đồ thị dòng chảy
  - Mỗi nút hình tròn biểu diễn một hoặc một vài tác vụ.
  - Cạnh có hướng miêu tả đường thực thi.
  - Đồ thị dòng chảy được xây dựng từ lưu đồ thuật giải.

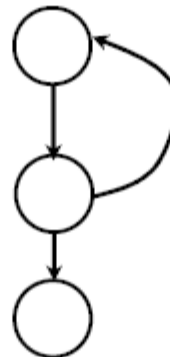
sequence



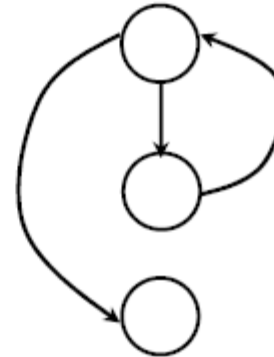
if



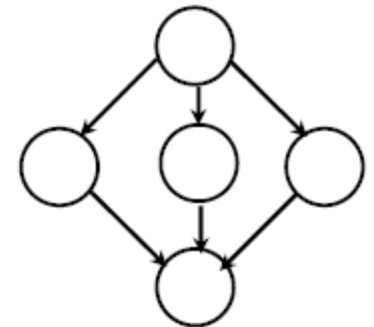
until



while



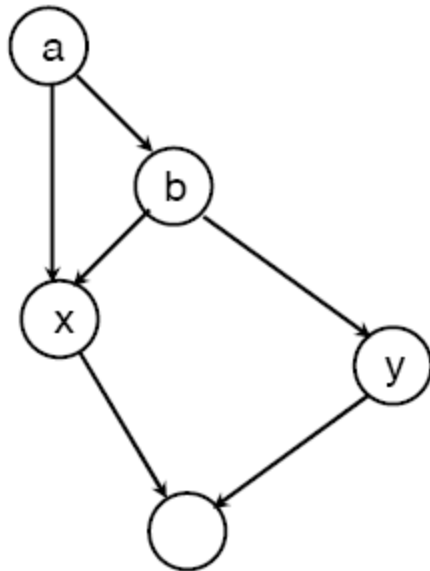
case



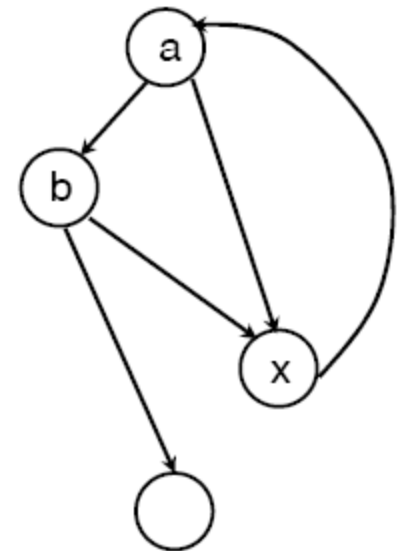
# Kiểm thử đường đi



- Thiết lập đồ thị dòng chảy
  - Phải phân rã tất cả các điều kiện phức trở thành các điều kiện đơn.



`if a and b then y else x`



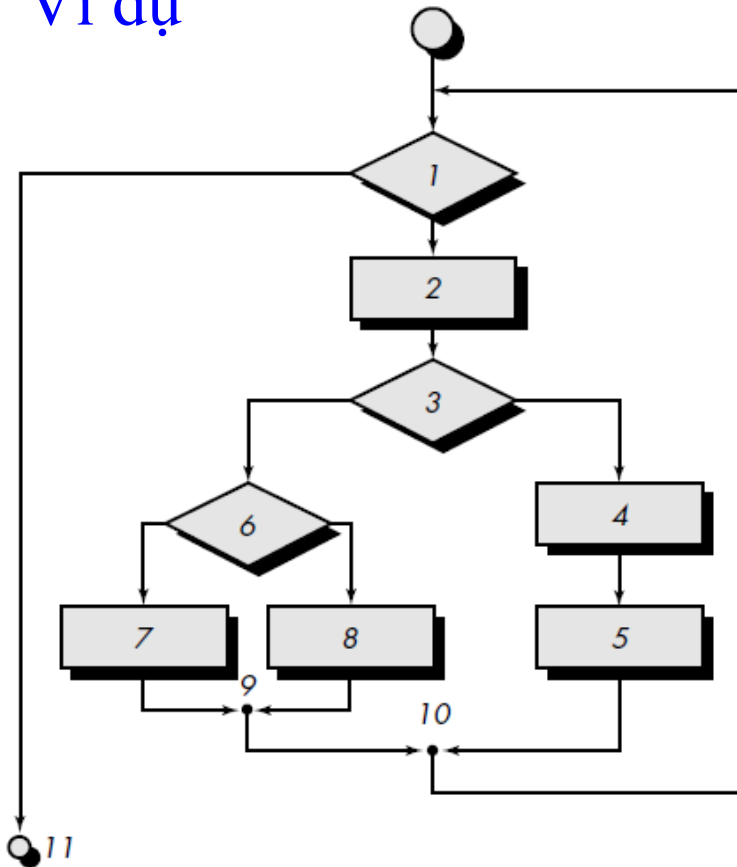
`while a or b do x`



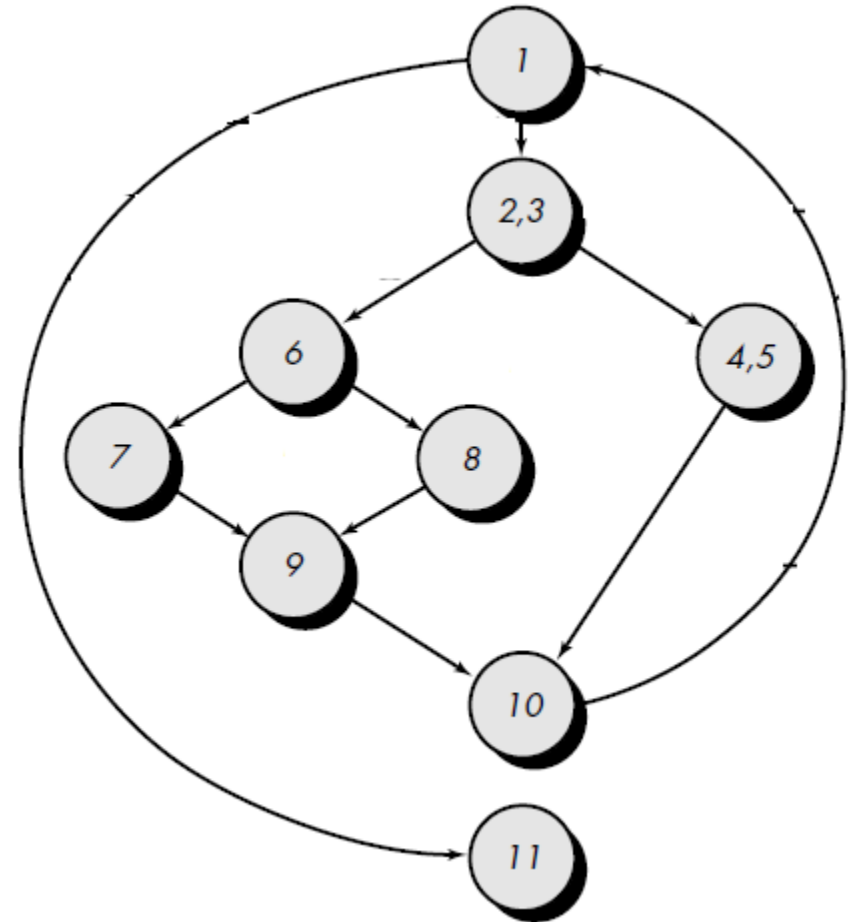
# Kiểm thử đường đi



- Ví dụ

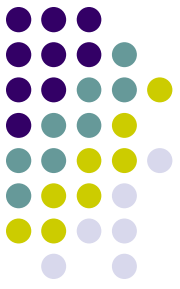


Lưu đồ giải thuật



Đồ thị dòng chảy

# Kiểm thử đường đi

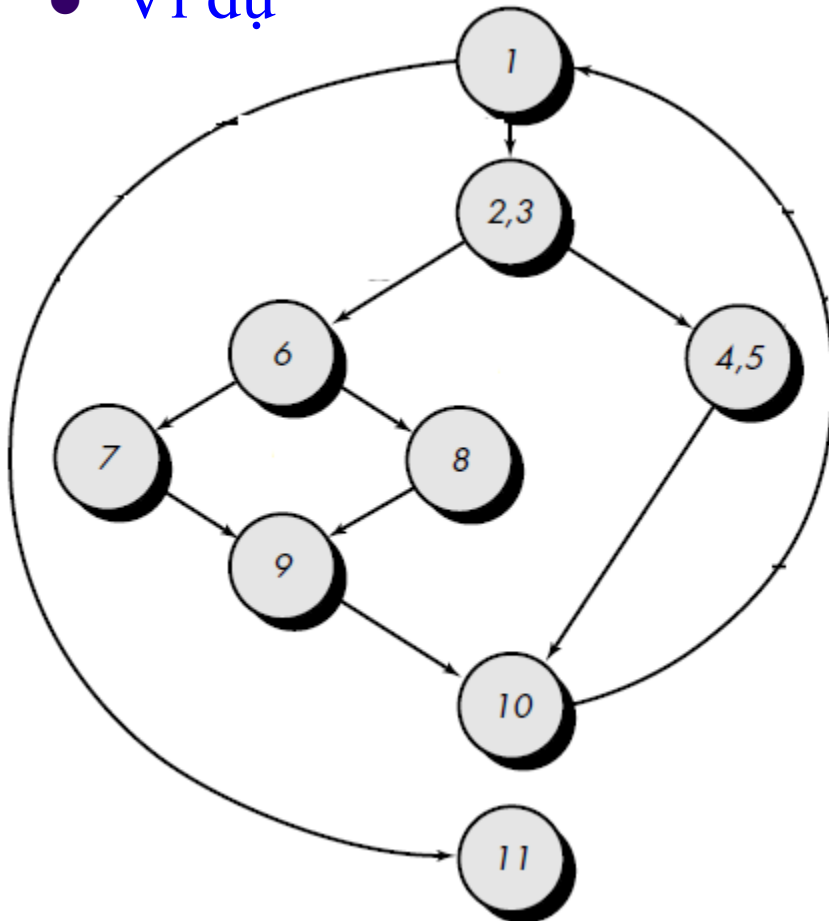


- Liệt kê các **đường độc lập cơ bản**
  - Từ nút bắt đầu đến nút kết thúc, các đường thực thi cơ bản được liệt kê theo một thứ tự nào đó để đảm bảo rằng đường đang liệt kê ít nhất đi qua một cạnh chưa được duyệt qua bởi các đường đã liệt kê trước đó .
  - Tổng số đường thực thi cơ bản độc lập nhau được tính bằng  **$V = P + 1$**  trong đó P là số nút phân nhánh.

# Kiểm thử đường đi



- Ví dụ



Đồ thị dòng chảy

Các đường độc lập cơ bản:

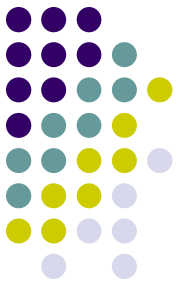
1. 1 - 11
2. 1 - 2,3 - 4,5 - 10 - 1 - 11
3. 1 - 2,3 - 6 - 7 - 9 - 10 - 1 - 11
4. 1 - 2,3 - 6 - 8 - 9 - 10 - 1 - 11

# Kiểm thử đường đi

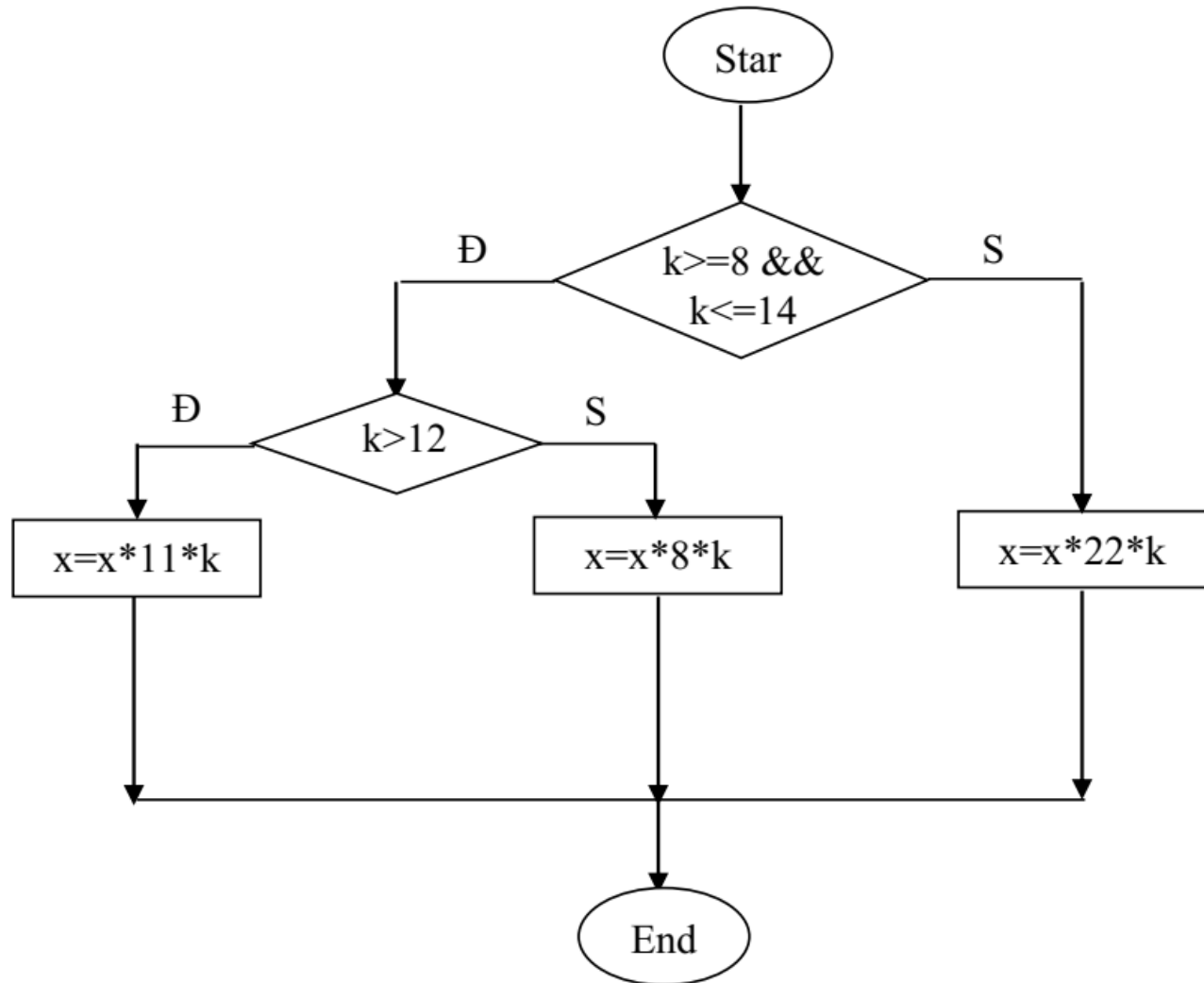


- Sinh ra **các trường hợp kiểm thử** (test-case)
  - Thiết lập một test-case cho mỗi đường thực thi cơ bản.
  - Dựa vào giải thuật để
    - Đưa ra dữ liệu input.
    - Tính ra dữ liệu output hay đáp ứng mong đợi của giải thuật.
  - Có thể không tạo ra được test-case cho một đường thực thi nào đó.

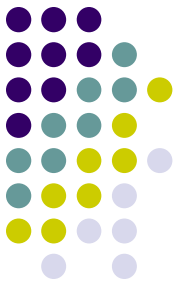
# Kiểm thử đường đi



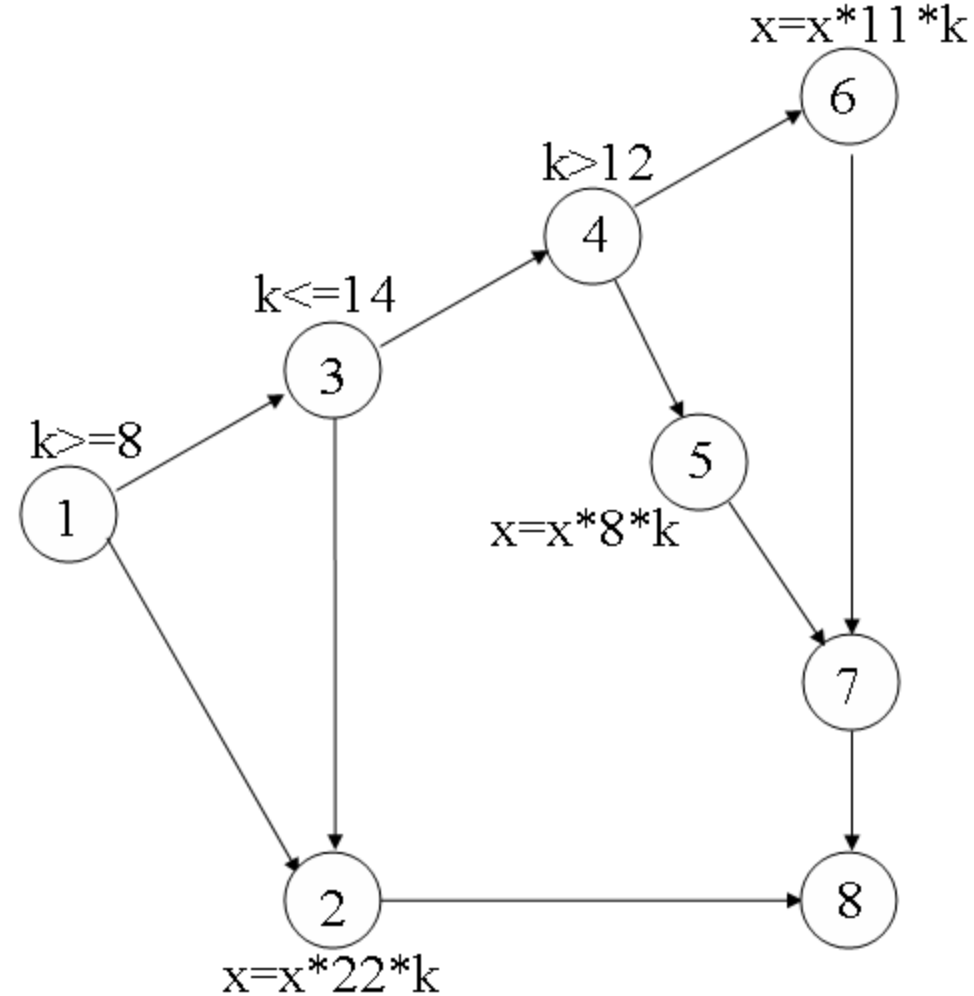
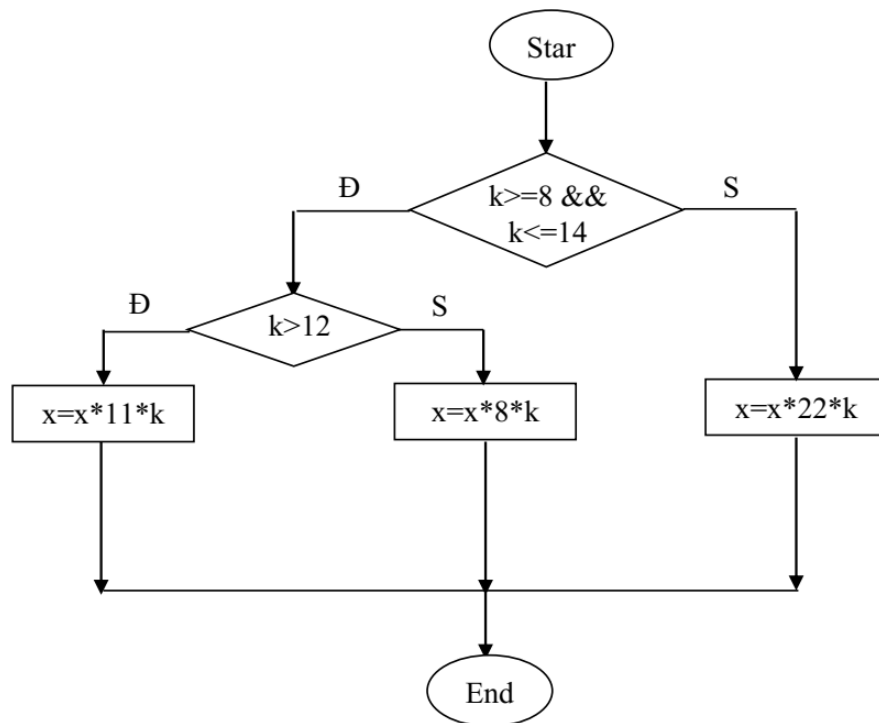
- **Ví dụ:** Cho lưu đồ sau, hãy xác định các trường hợp kiểm thử?



# Kiểm thử đường đi



- **Ví dụ:** Cho lưu đồ sau, hãy xác định các trường hợp kiểm thử?



# Kiểm thử đường đi

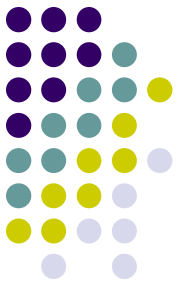


- Ví dụ (tt)

```
if (k >= 8 && k <= 14)    {  
    if (k > 12)  
        x = x * 11 * k;  
    else  
        x = x * 8 * k;    }  
else  
    x = x * 22 * k;
```

Test case	Input		Output mong đợi	Output thực tế	Kết luận
	x	k	x	x	
TH1	2	7	308	308	Thành công
TH2	3	15	990	900	Thất bại
TH3					
TH4					

# Kiểm thử tích hợp



- Kiểm thử tích hợp nhằm phát hiện các lỗi liên quan đến sự giao tiếp giữa các thành phần.
  - Các dạng kiểm thử tích hợp
    - Kiểm thử từ dưới lên
    - Kiểm thử từ trên xuống
    - Kiểm thử từ trên xuống được hiệu chỉnh
    - Kiểm thử sandwich (kẹp)
    - Kiểm thử sandwich được hiệu chỉnh
    - Kiểm thử Big-bang
- Tích hợp dần*
- Tích hợp đồng thời một lúc*



# Kiểm thử tích hợp

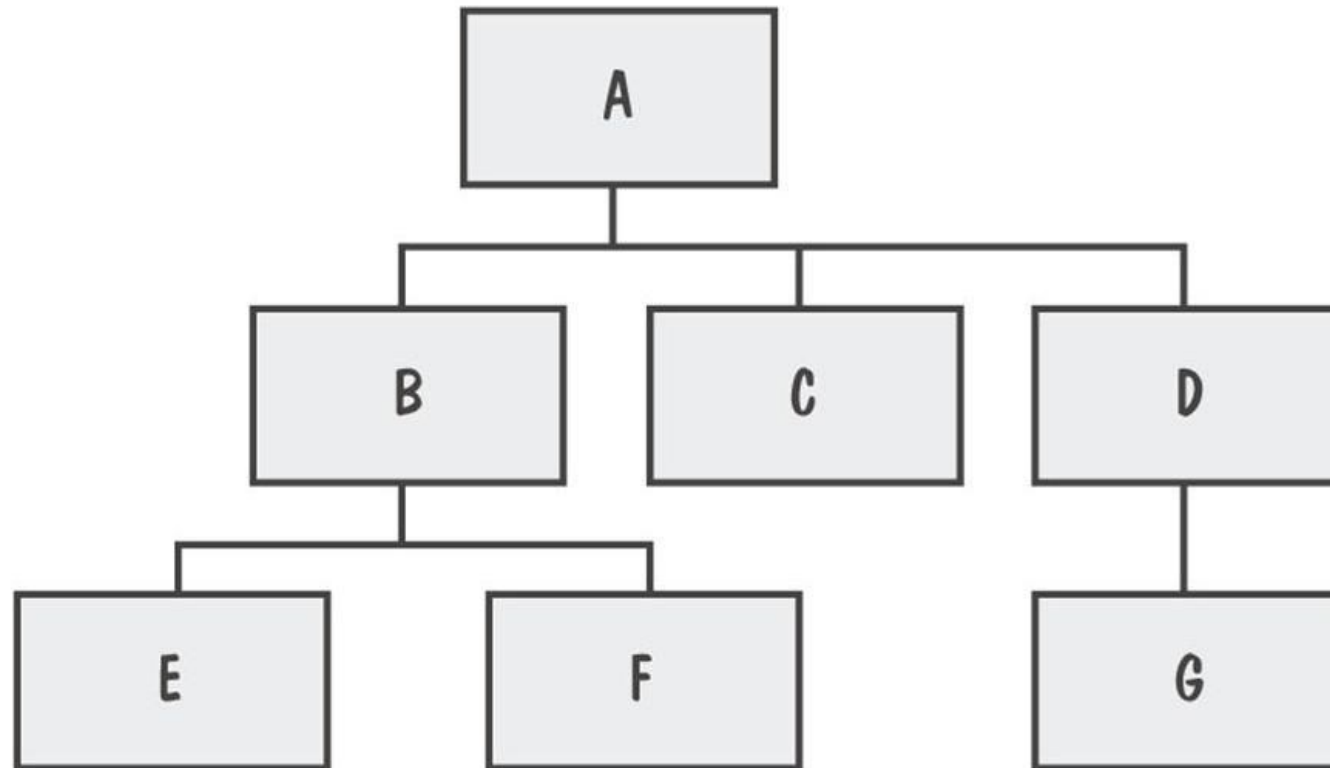


- Thuật ngữ
  - **Driver**: một chương trình mà nó gọi một thành phần (component) cụ thể và truyền một trường hợp kiểm thử cho nó.
  - **Stub**: một chương trình (có mục đích đặc biệt để) mô phỏng hoạt động của thành phần bị thiếu.



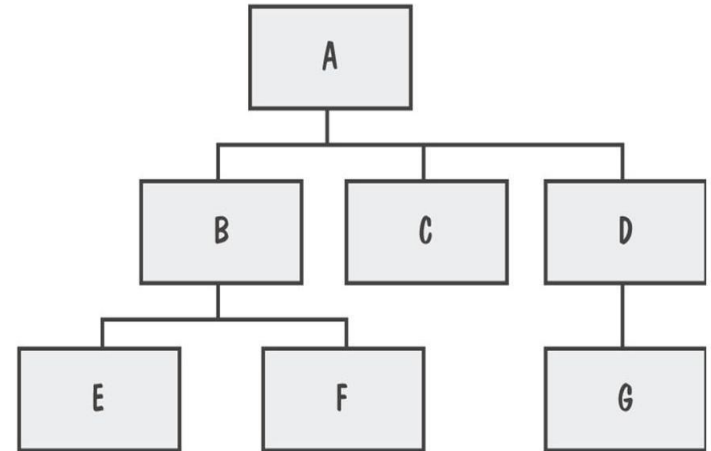
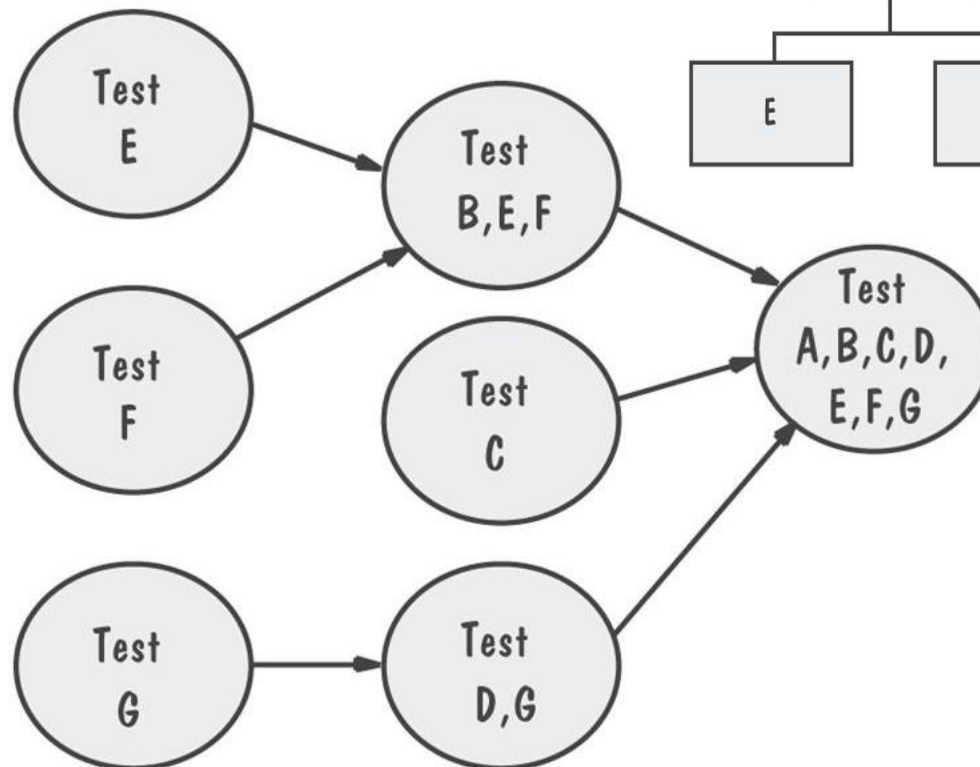
# Kiểm thử tích hợp

- Ví dụ: Cho một hệ thống với các thành phần được phân cấp



# Kiểm thử tích hợp

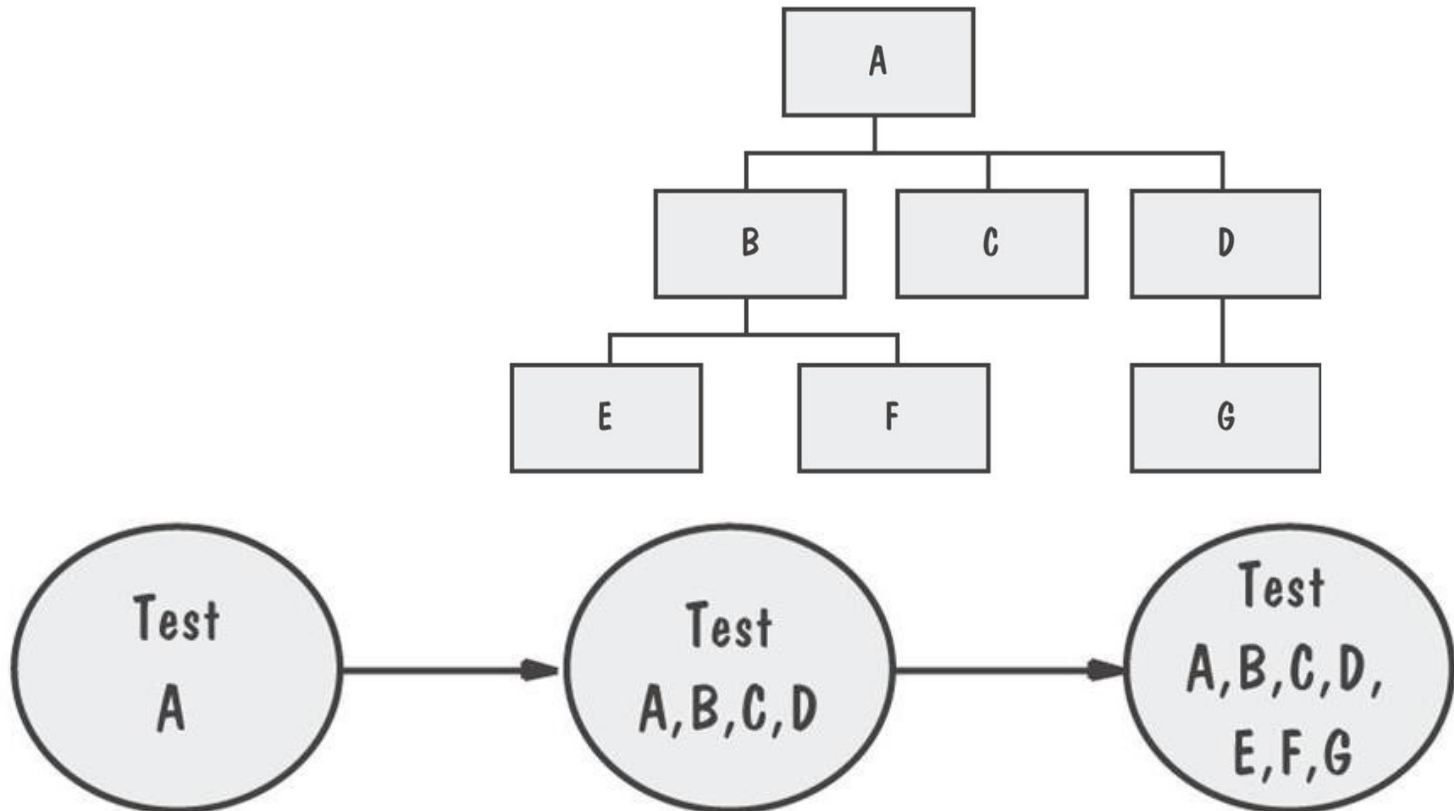
- Kiểm thử tích hợp từ dưới lên



# Kiểm thử tích hợp



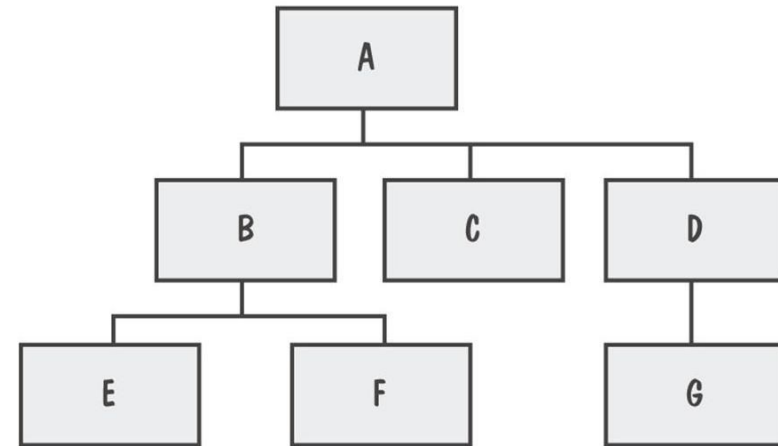
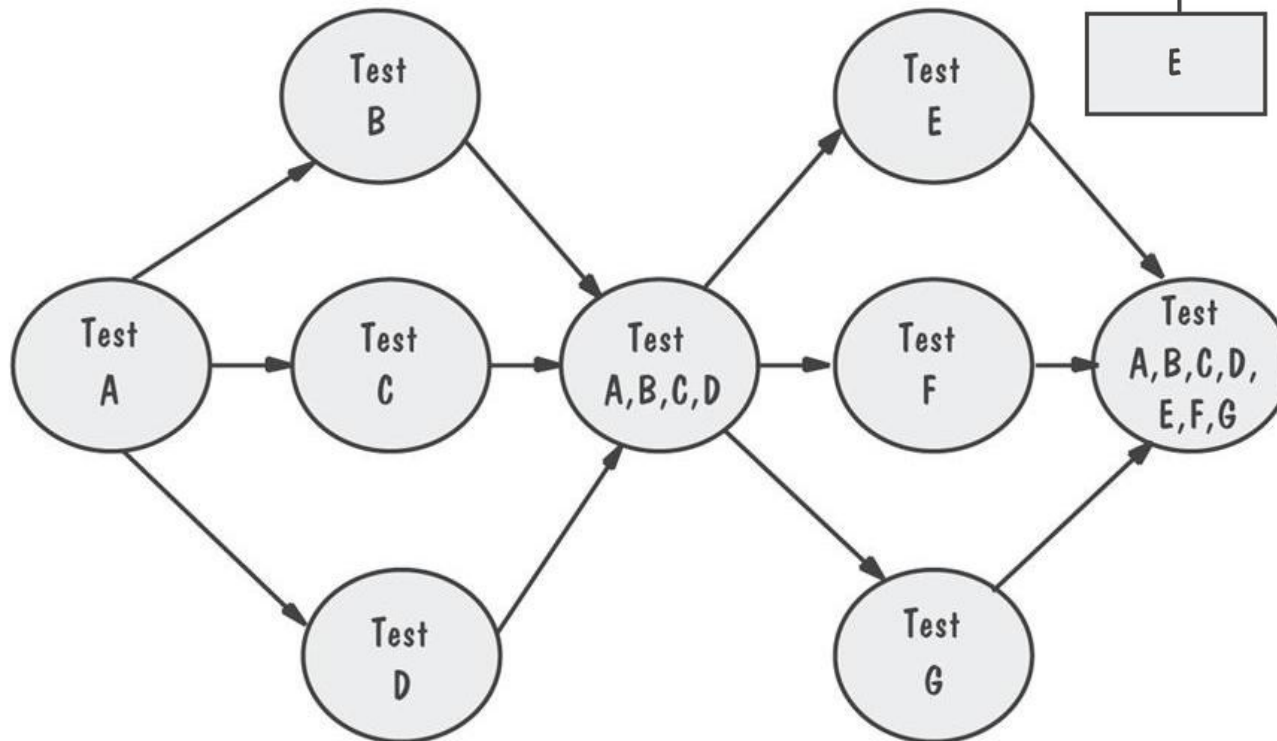
- Kiểm thử tích hợp từ trên xuống



# Kiểm thử tích hợp



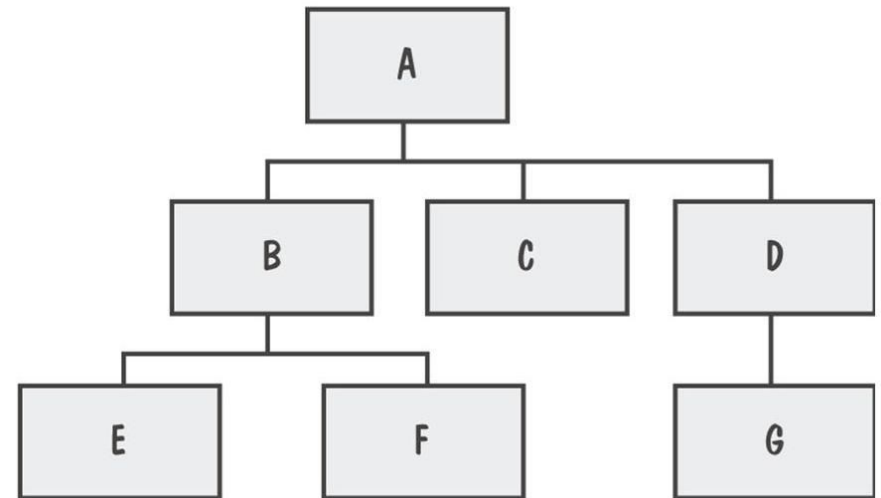
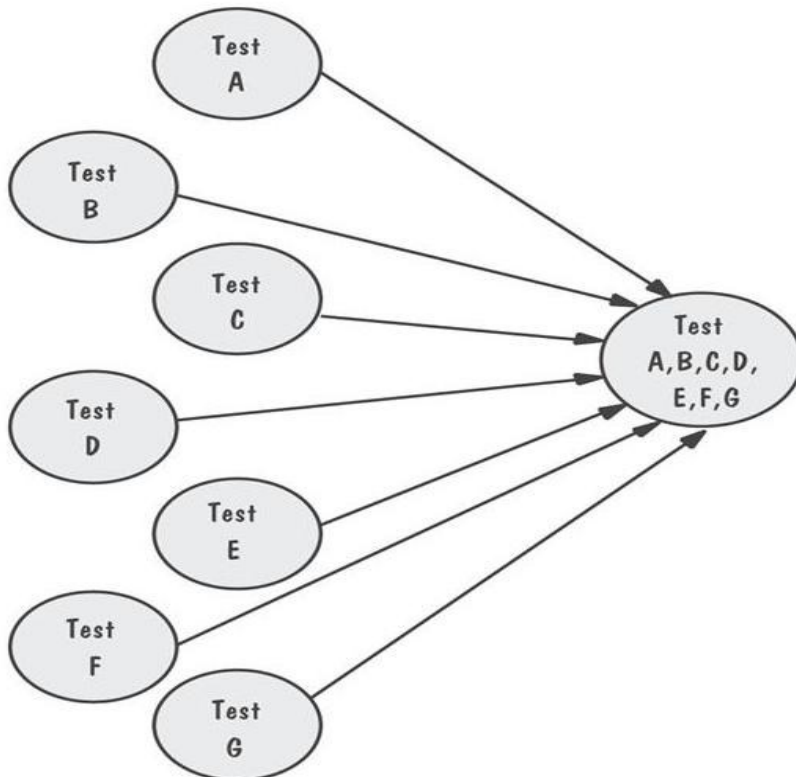
- Kiểm thử tích hợp từ trên xuống được hiệu chỉnh



# Kiểm thử tích hợp



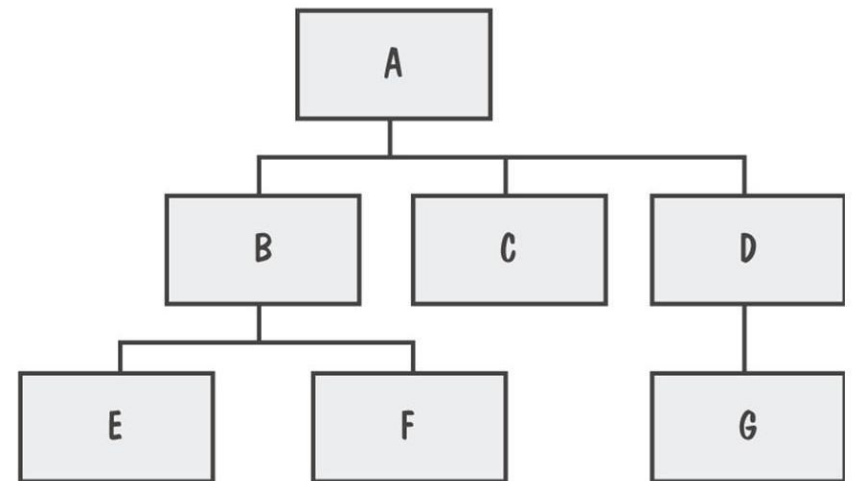
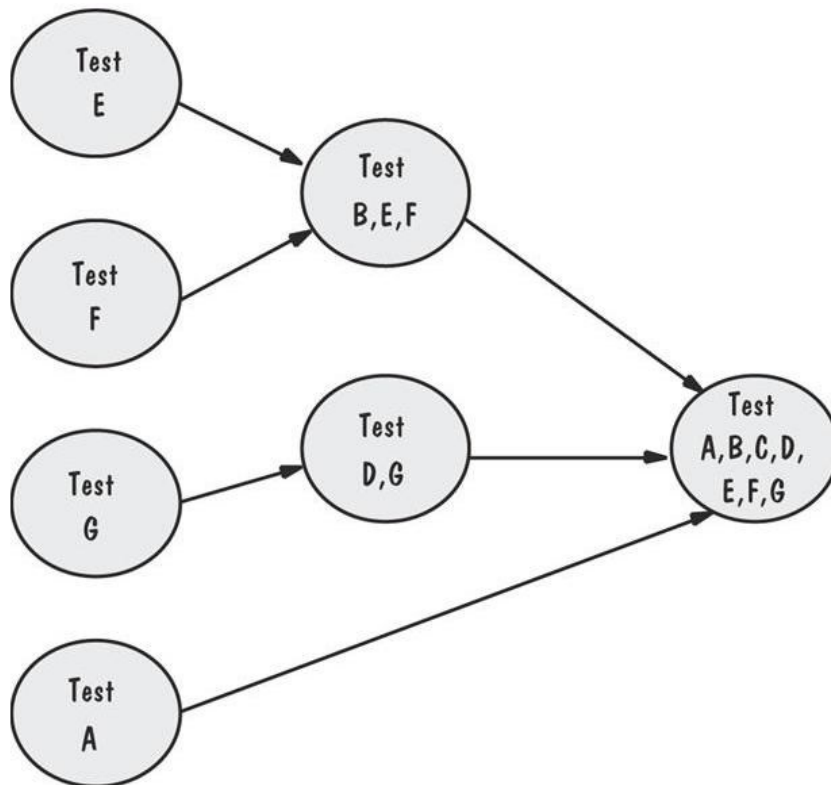
- Kiểm thử tích hợp Big Bang



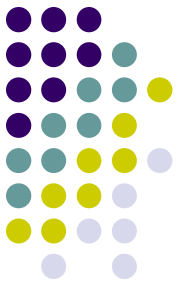
# Kiểm thử tích hợp



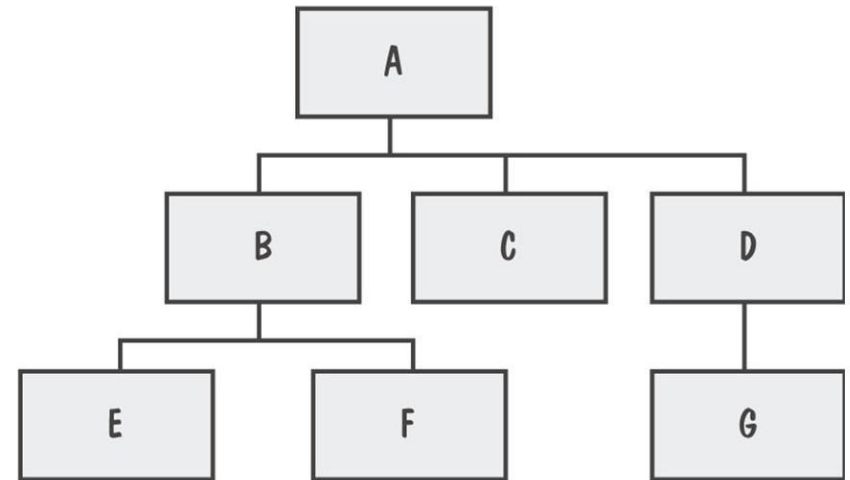
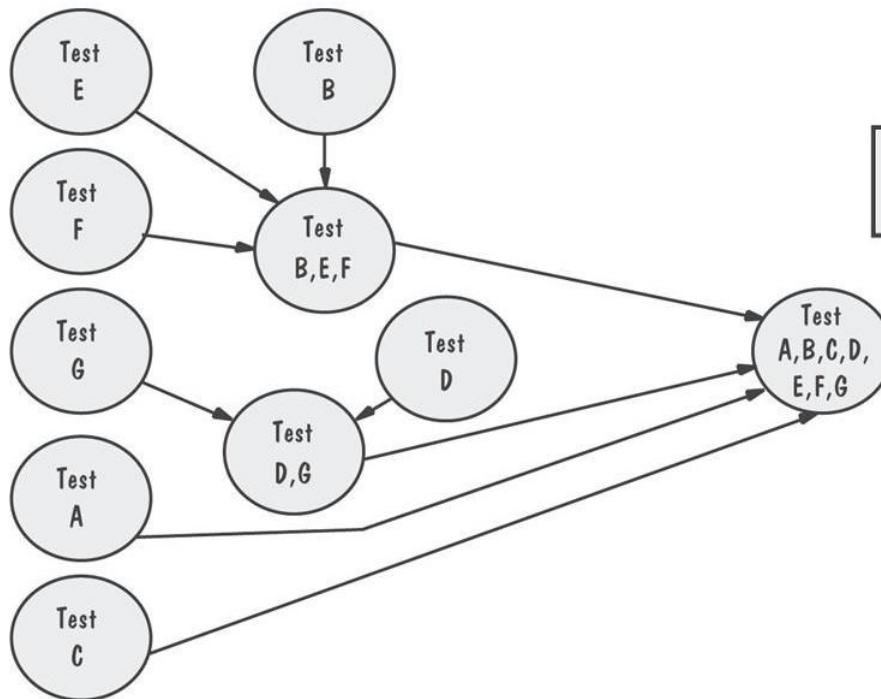
- Kiểm thử tích hợp sandwich



# Kiểm thử tích hợp



- Kiểm thử tích hợp sandwich được hiệu chỉnh





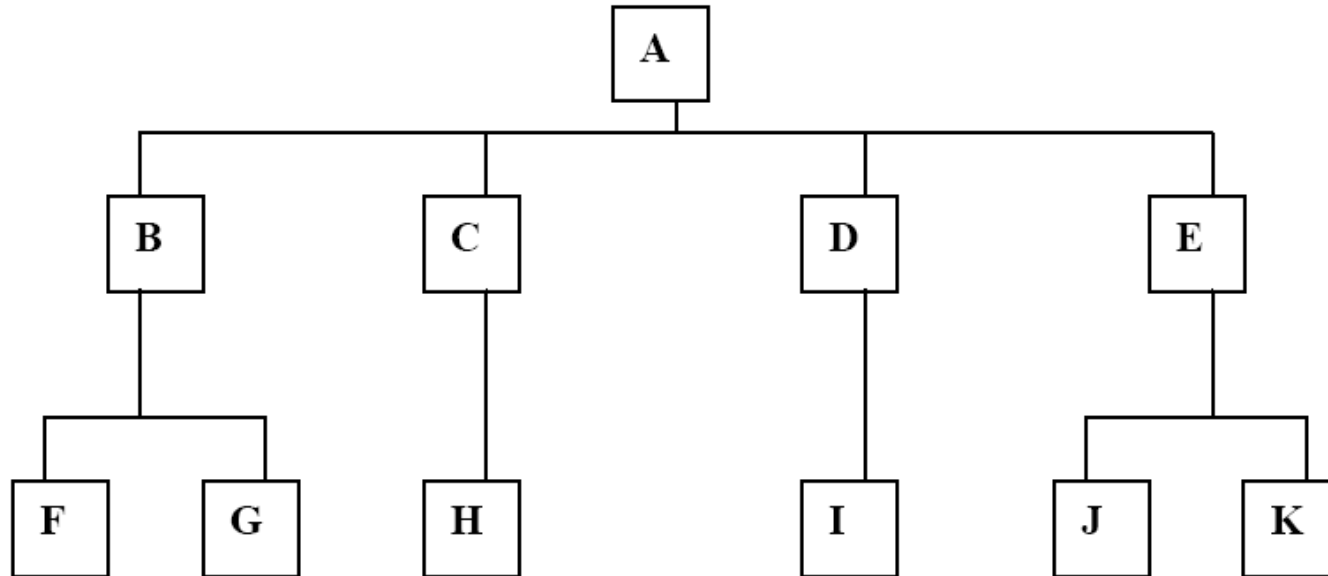
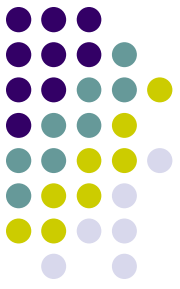
# Kiểm thử tích hợp



- So sánh các dạng kiểm thử tích hợp

	Từ dưới lên	Từ trên xuống	Từ trên xuống được hiệu chỉnh	Big-bang	Sandwich	Sandwich được hiệu chỉnh
Sự tích hợp	Early	Early	Early	Late	Early	Early
Thời gian cho lập trình công việc cơ bản	Late	Early	Early	Late	Early	Early
Các driver thành phần được cần đến	Yes	No	Yes	Yes	Yes	Yes
Các Stub được cần	No	Yes	Yes	Yes	Yes	Yes
Quan hệ song song của công việc lúc bắt đầu	Medium	Low	Medium	High	Medium	High
Khả năng kiểm thử các đường dẫn cụ thể	Easy	Hard	Easy	Easy	Medium	Easy
Khả năng lập kế hoạch và kiểm soát chuỗi sự kiện	Easy	Hard	Hard	Easy	Hard	hard

# Bài tập



Xác định dạng kiểm thử tích hợp?

# Lập kế hoạch kiểm thử



- Thiết lập các mục tiêu kiểm thử
- Thiết kế các trường hợp kiểm thử
- Viết các trường hợp kiểm thử
- Kiểm tra các trường hợp kiểm thử
- Thực thi các kiểm thử
- Đánh giá các kết quả kiểm thử



# Lập kế hoạch kiểm thử

- Mục đích của kế hoạch

Kế hoạch kiểm thử giải thích:

- Ai thực hiện kiểm thử
- Tại sao các kiểm thử được thực hiện
- Cách thức các kiểm thử được kiểm soát
- Khi nào các kiểm thử được thực hiện

# Lập kế hoạch kiểm thử



- Nội dung của kế hoạch:
  - Các mục tiêu của kiểm thử
  - Cách thực hiện kiểm thử
  - Chuẩn được sử dụng để xác định khi nào kiểm thử hoàn thành

# Công cụ kiểm thử tự động



- Công cụ thực hiện kiểm thử
  - Capture và replay
  - Các stub và driver
  - Môi trường kiểm thử tự động
- Bộ sinh các trường hợp kiểm thử



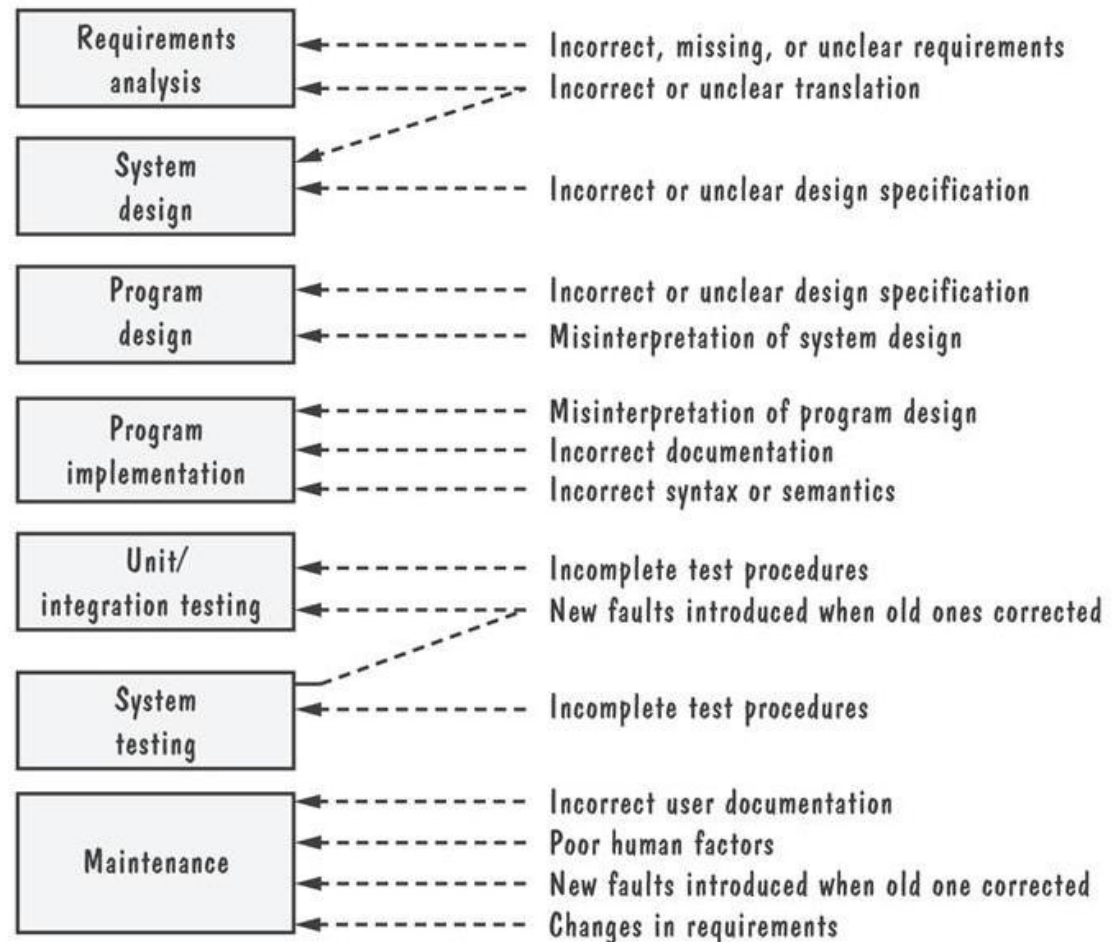
## Phần 2 - Nội dung

- Các nguyên lý của kiểm thử hệ thống
- Kiểm thử chức năng
- Kiểm thử sự thực thi
- Tính tin cậy, tính sẵn có và tính có thể bảo trì
- Kiểm thử chấp nhận
- Kiểm thử sự cài đặt
- Tài liệu kiểm thử

# Các nguyên lý của kiểm thử hệ thống



- Nguồn gốc của các lỗi phần mềm trong suốt quá trình phát triển





# Các nguyên lý của kiểm thử hệ thống

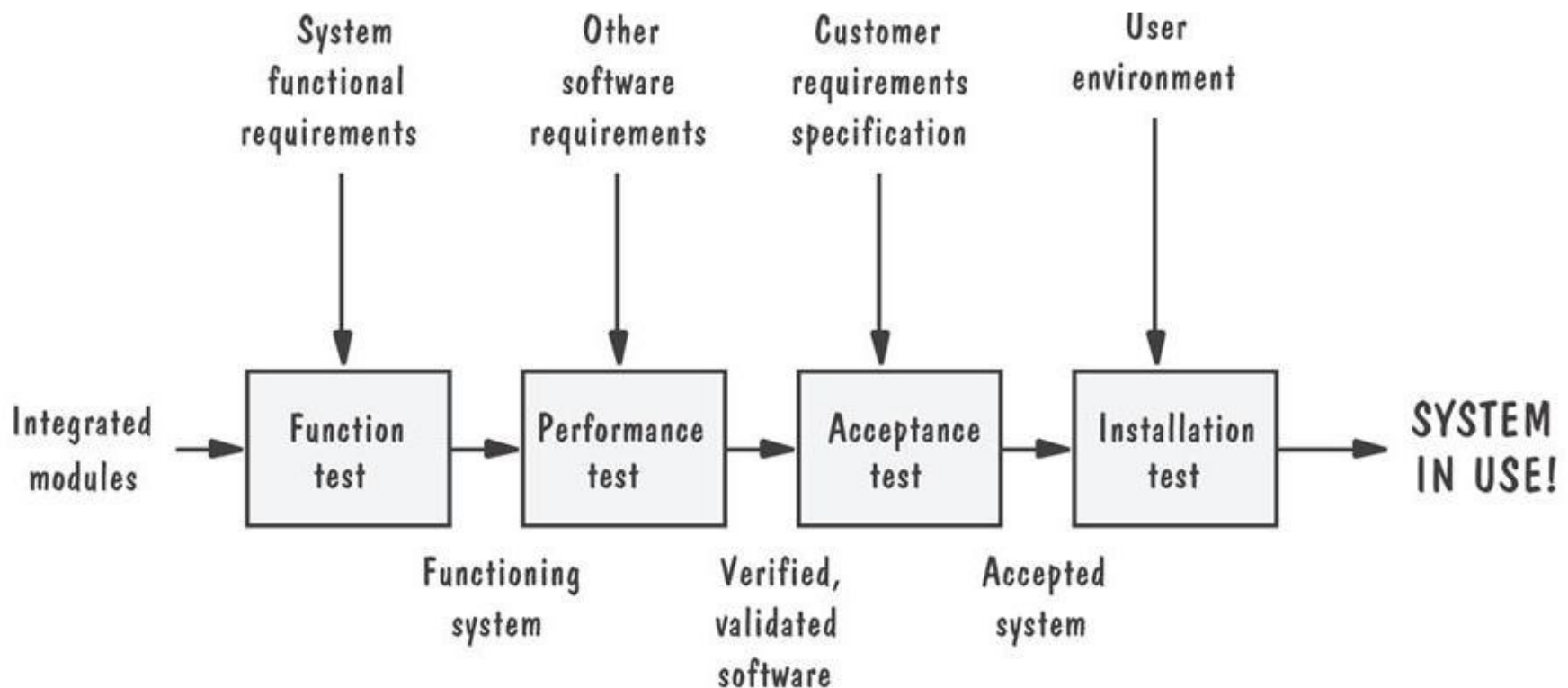


- Quy trình kiểm thử hệ thống
  - Kiểm thử chức năng: hệ thống được tích hợp có thực hiện như được cam kết trong đặc tả yêu cầu?
  - Kiểm thử thực hiện: các yếu tố phi chức năng có được đáp ứng?
  - Kiểm thử chấp nhận: hệ thống có phải là cái mà khách hàng mong đợi?
  - Kiểm thử sự cài đặt: hệ thống có vận hành ở chỗ khách hàng không?

# Các nguyên lý của kiểm thử hệ thống



- Các bước trong quy trình kiểm thử hệ thống



# Các nguyên lý của kiểm thử hệ thống



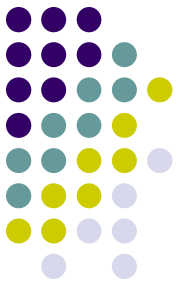
- Những kỹ thuật được sử dụng trong kiểm thử hệ thống
  - Kế hoạch tích hợp hay xây dựng.
  - Kiểm thử hồi quy.
  - Quản lý cấu hình.

# Các nguyên lý của kiểm thử hệ thống



- Kế hoạch tích hợp hay xây dựng
  - Mô tả các hệ thống con (spin) được kiểm thử.
  - Mô tả cách thức, nơi chốn, thời gian và người thực hiện các kiểm thử.

# Các nguyên lý của kiểm thử hệ thống



- Ví dụ về kế hoạch xây dựng cho hệ thống viễn thông

Spin	Chức năng	Bắt đầu kiểm thử	Kết thúc kiểm thử
0	Exchange	1 September	15 September
1	Area code	30 September	15 October
2	State/province/district	25 October	5 November
3	Country	10 November	20 November
4	International	1 December	15 December

# Các nguyên lý của kiểm thử hệ thống



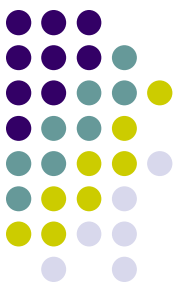
- Kiểm thử hồi quy
  - Nhận dạng các lỗi mới mà chúng được gây ra bởi các lỗi hiện tại được hiệu chỉnh.
  - Kiểm tra phiên bản hay phát hành mới để xác nhận rằng nó vẫn thực hiện cùng các chức năng theo cùng cách như phiên bản hay phát hành cũ.

# Các nguyên lý của kiểm thử hệ thống



- Các bước kiểm thử hồi quy
  - Chèn vào mã lệnh mới.
  - Kiểm thử các chức năng được biết sẽ bị ảnh hưởng bởi mã lệnh mới.
  - Kiểm thử các chức năng cần thiết của phiên bản  $m$  để xác nhận rằng chúng vẫn hoạt động chính xác.
  - Tiếp tục kiểm thử chức năng của phiên bản  $m + 1$ .

# Các nguyên lý của kiểm thử hệ thống



- Nhóm kiểm thử
  - Kiểm thử viên chuyên nghiệp: tổ chức và thực hiện các kiểm thử.
  - Nhà phân tích: người đã tạo ra các đặc tả.
  - Nhà thiết kế hệ thống: hiểu giải pháp được đề nghị.
  - Chuyên gia quản lý cấu hình: giúp kiểm soát các sửa chữa.
  - Người dùng: đánh giá các phát sinh.



# Kiểm thử chức năng



- Mục đích và vai trò
  - So sánh sự thực hiện thực tế của hệ thống với các yêu cầu của nó.
  - Phát triển các trường hợp kiểm thử dựa trên tài liệu yêu cầu.

# Kiểm thử chức năng



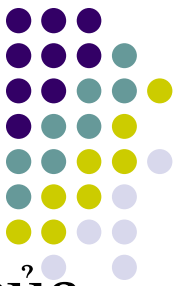
- Tạo ra các trường hợp kiểm thử chức năng
  - Vẽ đồ thị nhân quả từ các yêu cầu
  - Chuyển đồ thị thành bảng quyết định
  - Mỗi cột trong bảng quyết định tương ứng với một trường hợp kiểm thử chức năng

# Kiểm thử chức năng

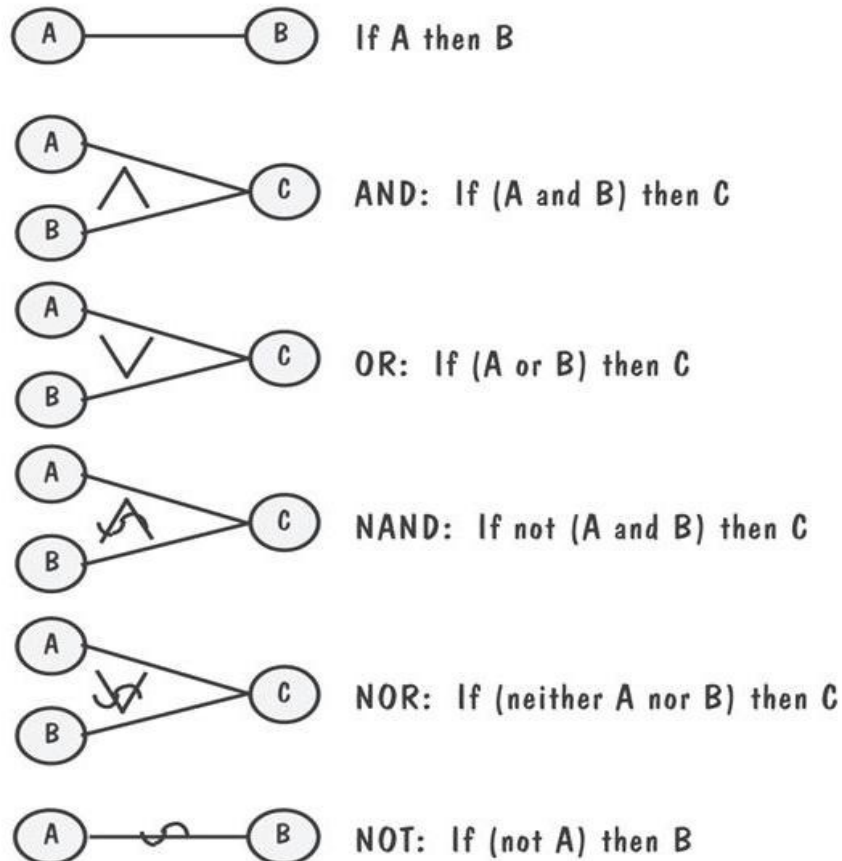


- Đồ thị nhân quả
  - Một đồ thị logic phản ánh quan hệ logic giữa nguyên nhân (các input) và kết quả (output hay sự biến đổi)
  - Vẽ đồ thị nhân quả từ các yêu cầu:
    - Các yêu cầu được phân tách sao cho mỗi yêu cầu mô tả một chức năng.
    - Mô tả các nguyên nhân, các kết quả (đó là các nút trong đồ thị với các nhân nằm bên trái và các quả nằm bên phải).
    - Vẽ mối quan hệ giữa chúng.

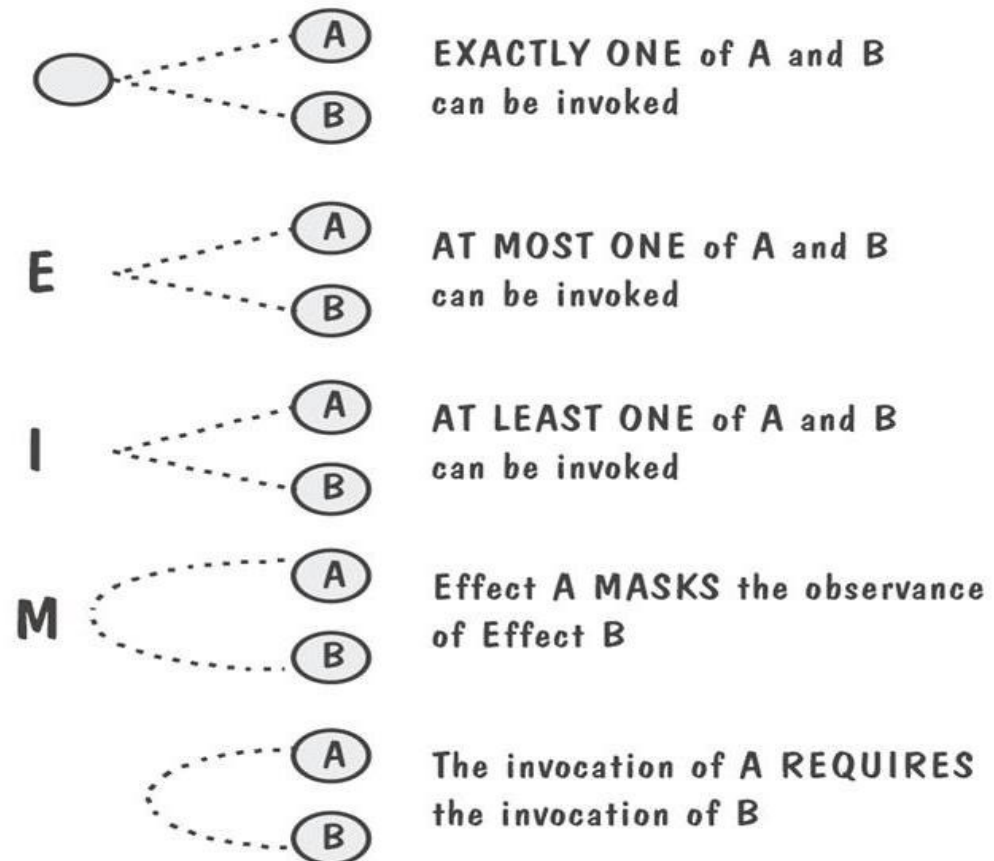
# Kiểm thử chức năng



- Ký hiệu của đồ thị nhân quả



- Ký hiệu bổ sung của đồ thị nhân quả



# Kiểm thử sự thực thi



- Mục đích và vai trò
  - Được sử dụng để kiểm tra:
    - Sự tính toán
    - Tốc độ đáp ứng
    - Độ chính xác của kết quả
    - Khả năng truy cập dữ liệu
  - Được thiết kế và quản lý bởi nhóm kiểm thử.

# Kiểm thử sự thực thi



- Các loại kiểm thử sự thực thi
  - Kiểm thử ứng suất
  - Kiểm thử dung lượng CTDL
  - Kiểm thử cấu hình
  - Kiểm thử tính tương thích
  - Kiểm thử hồi quy
  - Kiểm thử sự bảo mật
  - Kiểm thử sự điều hòa thời gian
  - Kiểm thử môi trường
  - Kiểm thử chất lượng
  - Kiểm thử sự hồi phục
  - Kiểm thử bảo trì
  - Kiểm thử tài liệu
  - Kiểm thử tính dễ sử dụng

# Tính tin cậy, tính sẵn có và tính có thể bảo trì



- Định nghĩa

- Tính tin cậy của phần mềm: vận hành mà không có lỗi dưới một điều kiện xác định trong một khoảng thời gian cho trước.
- Tính sẵn có của phần mềm: vận hành thành công theo sự đặc tả tại một điểm thời gian xác định.
- Tính có thể bảo trì của phần mềm: với một điều kiện sử dụng xác định, một hoạt động bảo trì có thể được thực hiện trong khoảng thời gian, thủ tục và tài nguyên xác định.

# Tính tin cậy, tính sẵn có và tính có thể bảo trì



- Các mức độ khác nhau về lỗi
  - Thảm khốc: gây ra sự chết chóc hoặc sự thất bại của hệ thống.
  - Then chốt: gây ra tổn hại rất xấu hay sự hư hại hệ thống chính dẫn đến sự thất bại về nhiệm vụ.
  - Lề: gây ra tổn hại phụ hay sự hư hại hệ thống phụ dẫn đến sự ngừng trệ, mất tính sẵn có hay giảm sút trong công việc.
  - Thứ yếu: không đủ nghiêm trọng để gây ra tổn hại hay sự hư hại hệ thống nhưng sẽ dẫn đến việc bảo trì và sửa chữa không có kế hoạch.



# Kiểm thử chấp nhận



- Mục đích và vai trò
  - Cho phép khách hàng và người dùng xác định xem hệ thống được xây dựng có đáp ứng được yêu cầu và sự mong đợi của họ hay không.
  - Được viết, quản lý và đánh giá bởi khách hàng.

# Kiểm thử chấp nhận



- Các loại kiểm thử chấp nhận
  - Kiểm thử thử nghiệm (pilot): cài đặt hệ thống trên cơ sở thực nghiệm.
    - Kiểm thử alpha: kiểm thử của người dùng trong tổ chức hay công ty phát triển phần mềm.
    - Kiểm thử beta: kiểm thử của khách hàng.
  - Kiểm thử song song: một hệ thống mới vận hành song song với hệ thống cũ.



# Kiểm thử sự cài đặt

- Trước khi kiểm thử
  - Cấu hình hệ thống
  - Gắn số và loại thiết bị
  - Thiết lập sự giao tiếp với hệ thống khác
- Kiểm thử
  - Kiểm thử hồi quy: kiểm tra rằng hệ thống được cài đặt một cách chính xác và hoạt động.



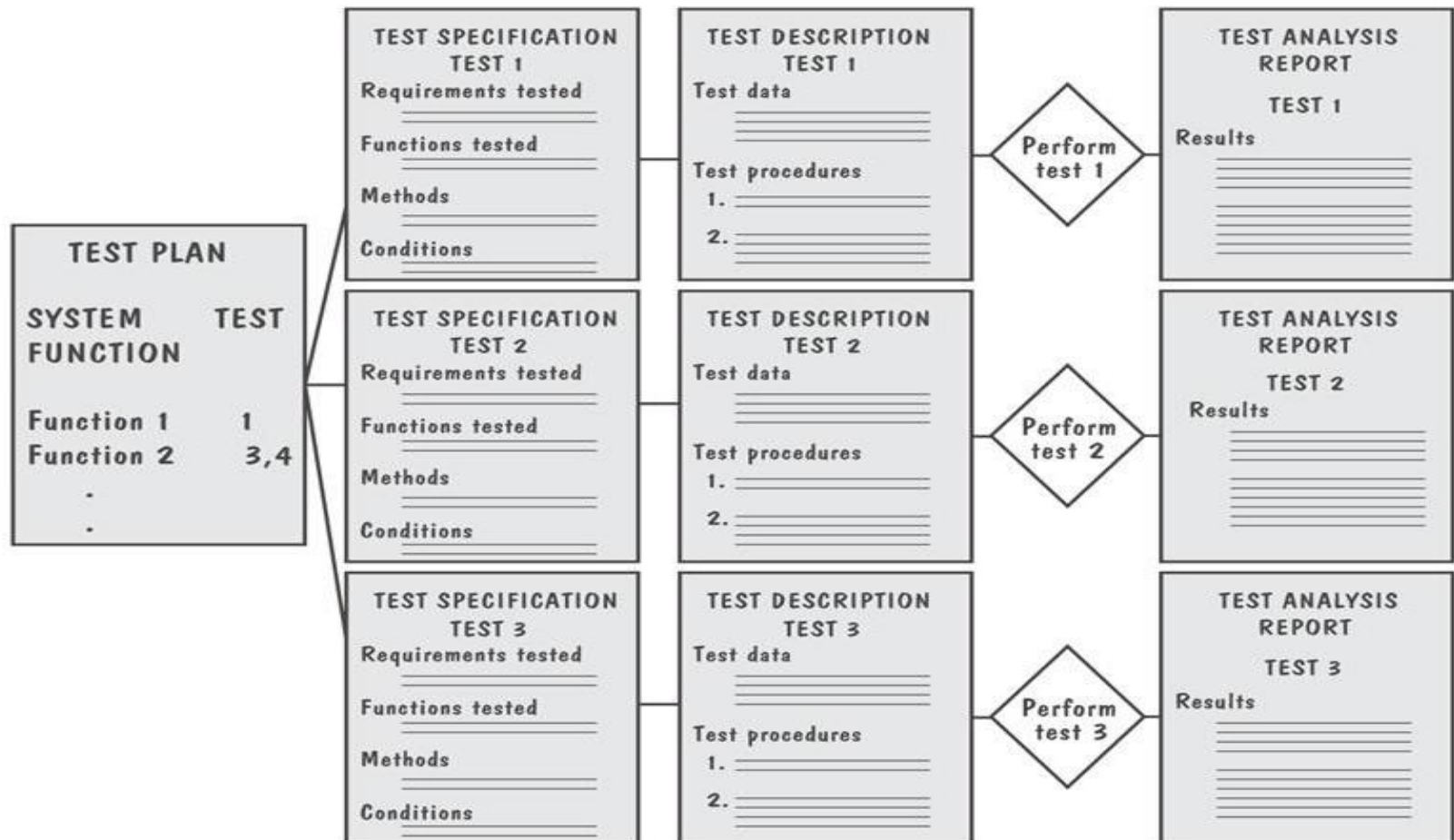
# Tài liệu kiểm thử

- Kế hoạch kiểm thử: mô tả hệ thống và kế hoạch để kiểm tra tất cả các chức năng và các đặc trưng.
- Đặc tả và đánh giá kiểm thử: chi tiết từng kiểm thử và định nghĩa tiêu chí để đánh giá từng đặc điểm.
- Mô tả kiểm thử: thủ tục và dữ liệu kiểm thử cho từng kiểm thử.
- Báo cáo phân tích kiểm thử: các kết quả của từng kiểm thử.

# Tài liệu kiểm thử



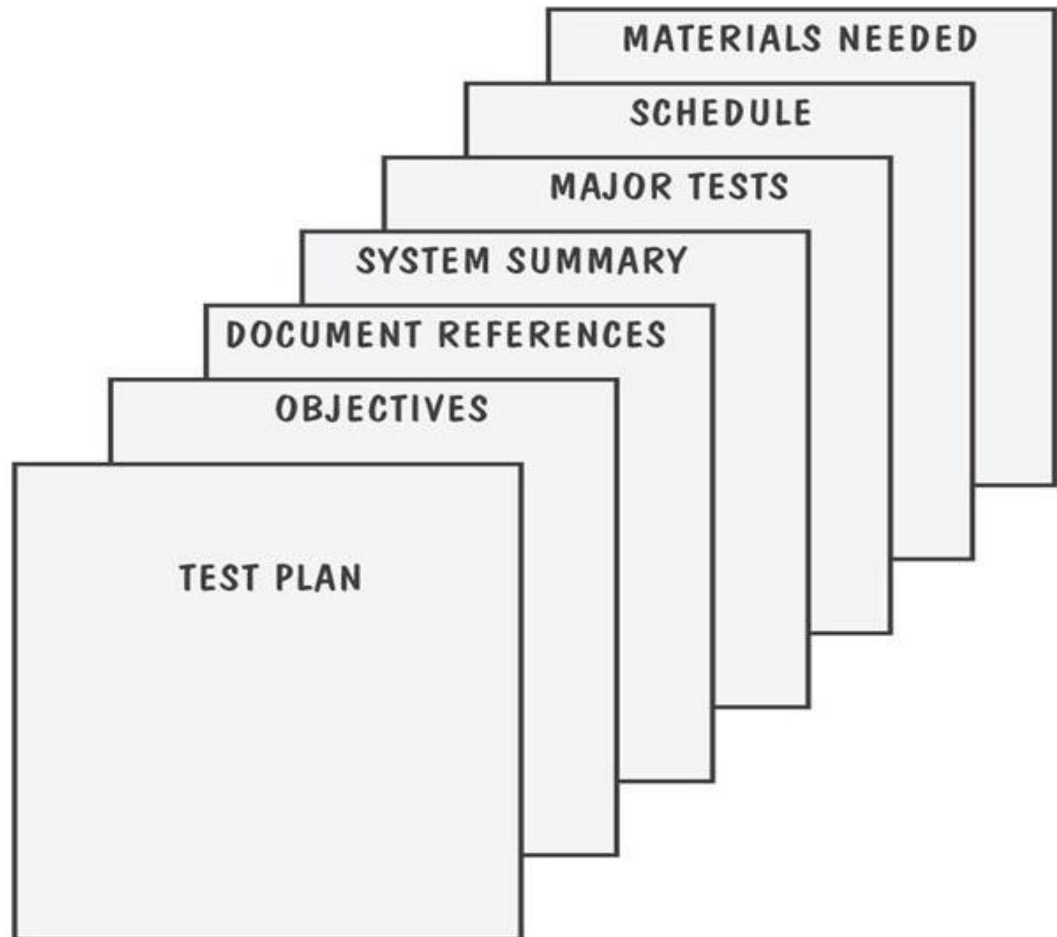
- Các tài liệu được tạo ra trong suốt sự kiểm thử



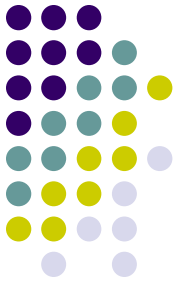
# Tài liệu kiểm thử



- Các phần của một kế hoạch kiểm thử



# Tài liệu kiểm thử



- Sự mô tả kiểm thử gồm:
  - Phương tiện kiểm soát
  - Dữ liệu
  - Thủ tục



# Tài liệu kiểm thử

- Báo cáo phân tích kiểm thử
  - Ghi kết quả kiểm thử.
  - Cung cấp thông tin cần để sao chép lại sự thất bại, định vị và sửa tận gốc vấn đề.
  - Cung cấp thông tin cần thiết để xác định xem dự án có hoàn thành hay không.
  - Thiết lập sự tin cậy trong sự thực thi của hệ thống.



# Tài liệu kiểm thử



- Biểu mẫu báo cáo vấn đề
  - Vị trí: vấn đề đã xuất hiện ở đâu?
  - Thời gian: nó đã xuất hiện khi nào?
  - Dấu hiệu: cái được quan sát?
  - Kết quả cuối cùng: các hệ quả?
  - Kỹ thuật: nó đã xuất hiện như thế nào?
  - Nguyên nhân: tại sao nó xuất hiện?
  - Tính khốc liệt: mức độ mà người dùng hay doanh nghiệp bị ảnh hưởng?
  - Chi phí: nó tốn bao nhiêu?



# HẾT PHẦN II.4