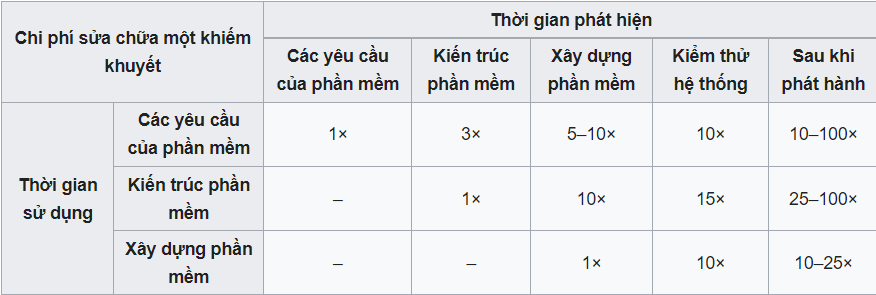
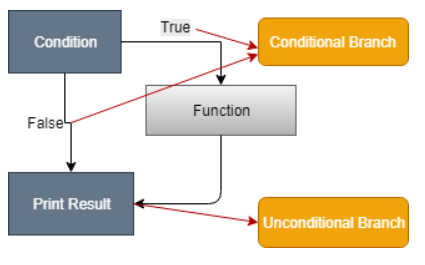
Họ và tên: Diệp Thị Linh

MSV : 2020600872

Bài Kiểm Tra Thường Xuyên Số 1

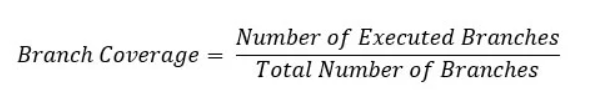
Câu 1:

* Kiểm thử phần mềm: là một cuộc kiểm tra được tiến hành để cung cấp cho các bên liên quan thông tin về chất lượng của sản phẩm hoặc dịch vụ kiểm thử.
  + Kiểm thử cung cấp cho doanh nghiệp một quan điểm, một cách nhìn độc lập về phần mềm để từ đó cho phép đánh giá và thấu hiểu những rủi ro trong quá trình triển khai phần mềm.
  + Kiểm thử phần mềm nhằm:
    - Đáp ứng được mọi yêu cầu hướng dẫn khi thiết kế và phát triển phần mềm.
    - Thực hiện công việc đúng như kỳ vọng.
    - Có thể triển khai được với những đặc tính tương tự.
    - Đáp ứng được mọi nhu cầu của các bên liên quan.
  + Mục đích chính của kiểm thử: là phát hiện các lỗi phần mềm để từ đó khắc phục và sửa chữa.
  + Phạm vi của kiểm thử thường bao gồm việc kiểm tra mã, thực hiện các mã trong môi trường va điều kiện khác nhau, việc kiểm thử các khía cạnh của mã: nó làm có đúng nhiệm vụ của nó hay không, nó có làm những gì cần phải là hay không.
* Kiểm thử phần mềm giúp tiết kiệm được chi phí.
  + Vidu: Một vấn đề được tìm thấy sau khi đã bán ra sản phẩm mềm chính thức sẽ chó chi phí gấp 10 -100 lần khi giải quyết vấn để từ lúc tiếp nhận yêu cầu.
  + Sự xuất hiện của cách thực triển khai thwujc tiến liên tục và các dịch vụ dựa trên đám mây, chi phí triển khai và bảo trì có thể giảm bớt theo thời gian.
  + 
* Lịch sử phát triển của Kiểm thử phần mềm:
  + Sự tách biệt giữa gỡ lỗi với kiểm thử lần đầu được Glenford J.Myers đưa ra vào năm 1979.
  + Năm 1988, Dave Gelperin và William C.Hetzel đã phân biệt các giai đoạn và mục tiêu trong kiểm thử phần mềm theo trình tự sau:
    - Trước 1956: Hướng về kiểm soát lỗi
    - 1957 – 1978: Hướng về chứng minh lỗi.
    - 1979 – 1982: Hướng về tính phá hủy của lỗi.
    - 1983 – 1987: Hướng về đánh giá lỗi.
    - 1988 – 2000: Hưỡng về việc phòng ngừa lỗi.
* Những rủi ro khi hệ thống không được kiểm thử:
  + Rủi ro sản phẩm:
    - Liên quan đến khả năng một sản phẩm công việc có thể không đáp ứng nhu cầu chính đáng của người dùng và các bạn liên quan.
    - Khi rủi ro sản phẩm liên quan đến đặc tính chất lượng củ thể của sản phẩm, rủi ro sản phẩm còn được gọi là rủi ro chất lượng. Ví dụ về rủi ro sản phẩm gồm:
      * Phần mềm không thực hiện được các chức nắng dự định theo đúng đặc tả kĩ thuật.
      * Phần mềm có thể không thực hiện các chức năng mong muốn theo nhu cầu của người dùng, khách hàng hoặc các bên liên quan.
      * Kiến trúc hệ thống không hỗ trợ đầy đủ một số các yêu cầu phi chức nắng.
      * Một tính toán cụ thể có thể được thực hiện nhưng không chính xác trong một số trường hợp.
      * Cấu trúc điều khiển vòng lặp có thể được mã hóa không chính xác.
      * Thời gian đáp ứng có thể khoogn đủ cho hệ thông xử lý giao dịch hiệu suất cao.
      * Phản hồi về trải nghiệm người dùng (UX) có thể không đáp ứng mong đợi của sản phẩm.
  + Rủi ro dự án
    - Liên quan đến các tình huống, nếu chúng xảy ra, có thể tác động tiêu cực đến khả năng đạt được các mục tiêu của dự án.
    - Ví dụ về rủi ro dự án bao gồm:
      * Project issues (các vấn đề của dự án) :
        + Sự chậm trễ có thể xảy ra trong qua trình bàn giao sản phẩm, hoàn thành nhiệm vụ hoặc thỏa mãn tiêu chí dừng hoặc định nghĩa về việc thực hiện.
        + Ước tính không chính xác, việc tái phân bỏ vốn cho các dự án ưu tiên cao hơn hoặc chi phí chung cho toàn tổ chức có thể dẫn đến việc tài trợ không đầy đủ.
        + Thay đổi muộn có thể dẫn đến việc làm lại đáng kể.
      * Organizational issues( các vấn đề về tổ chức):
        + Kỹ năng, đào tạo và nhân viên có thể không đủ.
        + Vấn đề nhân sự có thể gây ra xung đột.
        + Người dùng, nhân viên nghiệp vụ hoặc chuyên gia về chủ đề có thể không có sẵn do các ưu tiên nghiệp vụ mâu thuẫn.
      * Political isues (các vấn đề chính sách):
        + Người kiểm thử có thể không truyền đạt đầy đủ kết quả kiểm tra cho developer hoặc người kiểm tra có thể không theo dõi thông tin được tìm kiếm trong quá trình kiểm thử và đánh giá.
        + Có thể có thái độ không phù hợp đối với kì vọng của thử nghiệm.
      * Technical issues ( các vấn đề về kĩ thuật) :
        + Requirements không được định nghĩa đầy đủ.
        + Requirements có thể không được đáp ứng với các ràng buộc hiện có.
        + Thay đổ dữ liệu, kế hoạch thay đổi và công cụ hỗ trợ có thể bị chậm trễ.
        + Quản lý lỗi kém
      * Supplier issues ( Các vấn đề về nhà cung cấp) :
        + Bên thứ ba có thể không cung cấp sản phẩm hoặc dịch vụ cần thiết hoặc phá sản.
        + Các vấn đề về hợp đồng.
* Trình bày kĩ thuật bao phủ nhánh (Branch Convergae) :
  + Được sử dụng để bao phủ tất cả các nhánh của đồ thị luồng điều khiển.
  + Bao gồm tất cả các kết quả có thể xảy ra của mỗi điều kiện của điểm quyết định ít nhất một lần.
  + Kỹ thuật bao phủ nhánh là một kỹ thuật kiểm thử hộp trắng đảm bảo rằng mọi nhánh của mỗi điểm quyết định phải được thực thi.
  + Đồ thị luồng điều khiển để tính toán số lượng các nhánh

(Condition : điều kiện,Print result : in kết quả,

Function : hàm,Conditional Branch: điều kiện của nhánh, Unconditional Branch : nhánh không có điều kiện)

* + Cách tính toán phạm vi phủ sóng của chi nhánh
    - Phương pháp tìm đường
      * Số lượng đường dẫn của các nhánh được thực thi được sử dụng để tính toán cùng phủ của Nhánh.
      * Công thức tính phạm vi bảo hiểm chi nhánh:



(Branch Coverage : Phạm vi chi nhánh

Number of Executed Branches : Số lượng chi nhánh đã được thực hiện

Total Number of Branches: Tổng số chi nhánh)

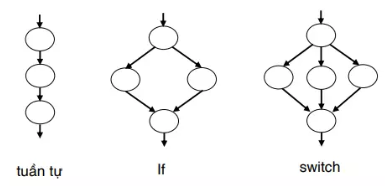
* + - * Ưu điểm của phạm vi chi nhánh:
        + Cho phép bạn xác thực tất cả các nhánh có trong mã.
        + Giúp bạn đảm bảo rằng không có nhánh dẫn đến bất kì sự bất thường nào trong hoạt động của chương trình.
        + Phương pháp phạm vi nhánh loại bỏ các sự cố xảy ra do kiểm tra phạm vi báo cáo
        + Cho phép bạn tìm những khu vực không được kiểm tra bằng các phương pháp kiểm tra khác.
        + Nó cho phép bạn tìm một thước đo định lượng về độ phủ của mã.
        + Vũng bao phủ nhánh bỏ qua các nhánh bên trong biểu thức Boolean.

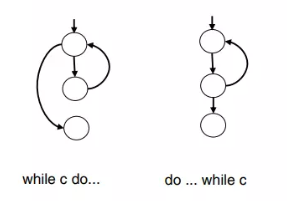
Câu 2:

* Kiểm thử hộp trắng trong các ứng dụng/ dịch vụ web:
  + Khái niệm:
    - Kiểm thử hộp trắng là một phương pháp kiểm thử phần mềm trong đó tester biết về cấu trúc nội bộ/ thiết kế. Người kiểm tra chọn đầu vào để thực hiện các đường dẫn thông qua mã và xác định đầu ra thích hợp.
    - Kiểm thử hộp trắng bao gồm phân tích dòng dữ liệu, điều khiển dòng, dòng thông tin, mã thực hành, ngoại lệ và những lỗi trình bày trong hệ thống để kiểm tra những hành động của phần mềm không được định hướng trước.
  + Đối tượng áp dụng:
    - Có thể là một thành phần phần mềm có thể là một hàm chức năng, một module chức năng, một phân hệ chức năng...
  + Mức độ áp ụng:
    - Unit Testing : Kiểm tra đường dẫn trong một đơn vị.
    - Integration Testing : Kiểm tra đường dẫn giữa các đơn vị.
    - System Testing : Kiểm tra các đường dẫn giữa các hệ thống con.
  + Ưu điểm của kiểm thử hộp trắng:
    - Test có thể bắt đầu ở giai đoạn sớm hơn, không cần phải chờ đợi cho GUI để có thể test.
    - Test kỹ càng hơn, có thể bao phủ hầu hết các đường dẫn.
    - Thích hợp trong việc tìm kiếm lỗi và các vấn đề trong mã lệnh.
    - Cho phép tìm kiếm các lỗi ẩn bên trong.
    - Các lập trình viên có thể tự kiểm tra.
  + Nhược điểm của kiểm thử hộp trắng:
    - Vì các bài kiểm thử rất phức tạp, đòi hỏi phải có các nguồn lực có tay nghề cao, với kiến thức sâu rộng về lập trình và thực hiện.
    - Maintenance test script có thể là một gánh nặng nếu thể hiện thay đổi quá thường xuyên.
    - Vì phương pháp thử nghiềm này liên quan chặt chẽ với ứng dụng đang được test, nên các công cụ để phục vụ cho mọi loại triển khai/ nền tắng có thể không sẵn có.
  + Các kỹ thuật kiểm thử hộp trắng phổ biến
    - Kiểm thử đường cơ bản – Đồ thị dòng
      * Đồ thị dòng gần giống đồ thị luồng điều khiển của chương trình.
      * Là phương pháp miêu tả thuật giải. Là phương pháp trực quan cho chúng ta thấy dễ dàng các thành phần của thuật giải và mối quan.
      * Có thể giúp những người thiết kế ca kiểm thử nhận được một độ phức tạp của 1 logic thủ tục.
      * Gồm 2 thành phần : các nút và các cung kết nối giữa chúng.
      * Các loại nút trong đồ thị dòng điều khiển:



* + - * Các kiểu cấu trúc thành phần đồ thị dòng:





* + - Kiểm thự dựa trên luồng điều khiển
      * Là một kịch bản thi hành đơn vị phần mềm tương ứng, cụ thể nó là danh sách có thứ tự các lệnh được thi hành ứng với 1 lần chạy cụ thể của đơn vị phần mềm, bắt đầu từ điểm nhập của đơn vị phần mềm đến điểm kết thúc của đơn vị phần mềm.
      * Có từ 1 đến n đường thi hành khác nhau.
      * Mục tiêu: đảm bảo mọi đường thi hành của đơn vị phần mềm cần kiểm thử đều chạy đúng.
    - Kiểm thử dựa trên luồng dữ liệu
    - Kiểm thử đột biến