

ÔN TẬP KIỂM ĐỊNH PHẦN MỀM

NGUYỄN VĂN TÍN

LÝ THUYẾT KIỂM ĐỊNH PHẦN MỀM

Câu 1

Kiểm thử hộp trắng (White box testing) dùng để kiểm tra cấu trúc. Dựa trên quá trình thực hiện xây dựng phần mềm. Tiêu chuẩn: Bao phủ dòng lệnh, bao phủ nhanh, bao phủ đường.

Ví dụ: Với hệ thống đăng ký tín chỉ thì kiểm tra chức năng đăng ký học phần xem các hoạt động đúng hay không nếu như chưa chọn lớp học phần hoặc chưa chọn “Tôi đồng ý”, hay là lớp đã bị đầy hay chưa, đã đến thời hạn đăng ký hay chưa. Và xem có hoạt động có đúng như code của chương trình hay không ?

Kiểm thử hộp đen (Back box testing) dùng kiểm tra chức năng. Dựa vào bản mô tả chức năng của chương trình, xem chương trình có thực sự cấp đúng chức năng đã mô tả trong bản chức năng hay không. Các trường hợp thử nghiệm (test case) sẽ được tạo ra dựa nhiều vào bản mô tả chức năng.

Ví dụ: Một hệ thống giao tác được yêu cầu thực thi 20 giao tác / giây. Khi đó sẽ kiểm tra nếu 30 giao tác / giây thì như thế nào ?

Kiểm thử module (Module testing): kiểm tra một bộ phận riêng rẽ. Một phương pháp theo cách tiếp cận giảm thiểu sót về 0 là áp dụng suy diễn toán học cho đòi hỏi logic, chứng minh tính đúng đắn của chương trình. Phương pháp này đòi hỏi đặc tả ngôn ngữ dạng hình thức để có thể chứng minh tính đúng đắn của chương trình thông qua các dòng lệnh đã viết.

Ví dụ: Với hệ thống xử lý văn bản thì kiểm tra các chức năng *tạo tài liệu, sửa tài liệu, xóa tài liệu...* có hoạt động đúng hoặc có lỗi hay không.

Tiêu chuẩn	Black box test	White box test
Định nghĩa	Kiểm tra hộp đen là phương pháp thử nghiệm phần mềm được sử dụng để kiểm tra các phần mềm mà không quan tâm đến cấu trúc bên trong của chương trình.	Kiểm tra hộp trắng là phương pháp kiểm thử phần mềm, sử dụng để kiểm tra phần mềm mà yêu cầu phải biết cấu trúc bên trong của chương trình.
Trách nhiệm	Thử nghiệm được thực hiện bên ngoài, không liên quan đến nhà phát triển phần mềm.	Thông thường, các thử nghiệm được thực hiện bởi nhà phát triển phần mềm.
Cấp độ test sử dụng	Thử nghiệm áp dụng ở cấp độ cao như: kiểm tra hệ thống (System test), kiểm tra chấp nhận (Acceptance test)	Thử nghiệm được áp dụng ở mức độ thấp hơn như thử nghiệm đơn vị (Unit Test), thử nghiệm hội nhập
Biết lập trình	Không yêu cầu hiểu biết về Lập trình	Yêu cầu hiểu biết nhất định về LT.
Biết việc thực hiện chương trình	Không yêu cầu hiểu về cấu trúc bên trong chức năng, và không cần hiểu làm thế nào để có được chức năng đó	Yêu cầu hiểu cấu trúc bên trong chức năng được thực hiện như nào.
Cơ sở tạo Test Cases	Kiểm tra hộp đen được bắt đầu dựa trên tài liệu yêu cầu kỹ thuật	Kiểm tra hộp trắng được bắt đầu dựa trên các tài liệu thiết kế chi tiết

Nội dung	Black box test	White box test
Ưu điểm	<ul style="list-style-type: none"> - Thích hợp trong việc kiểm tra từng phân đoạn lớn các mã lệnh, chức năng lớn - Người thử nghiệm không cần hiểu biết về mã lệnh được viết trong chương trình - Tách biệt giữa quan điểm của người sử dụng và người phát triển phần mềm 	<ul style="list-style-type: none"> - Thích hợp trong việc tìm kiếm lỗi và các vấn đề trong mã lệnh - Biết được yêu cầu bên trong của phần mềm, kiểm tra sẽ sát hơn - Cho phép tìm kiếm các lỗi ẩn bên trong - Các lập trình viên có thể tự kiểm tra - Giúp tối ưu việc mã hoá - Do yêu cầu kiến thức cấu trúc bên trong của phần mềm, nên việc kiểm soát lỗi tối đa nhất
Nhược điểm	<ul style="list-style-type: none"> - Độ bao phủ hạn chế vì chỉ có một phần nhỏ trong số các kịch bản thử nghiệm được thực hiện - Kiểm tra không hiệu quả do người thử nghiệm không hiểu biết gì về cấu trúc bên trong phần mềm. - Tester có hạn chế về hiểu biết về ứng dụng 	<ul style="list-style-type: none"> - Không thể tìm thấy tính năng chưa thực hiện hoặc bỏ sót - Đòi hỏi hiểu sâu về cấu trúc bên trong của phần mềm được thử nghiệm - Yêu cầu truy xuất mã lệnh bên trong chương trình

Câu 2: Khái niệm kiểm thử bằng phương pháp đồ thị nguyên nhân kết quả.

Cách thức để xác định các ca kiểm thử của phương pháp này.

Đồ thị nguyên nhân-kết quả hỗ trợ cho việc lựa chọn một cách có hệ thống tập các ca

kiểm thử có hiệu quả cao. Nó tác động tới việc chỉ ra các tình trạng chưa đầy đủ hoặc nhập

những trong đặc tả. Ngoài ra nó còn cung cấp cách biểu diễn chính xác cho các điều kiện logic

và hành động tương ứng.

Các bước được sử dụng để xây dựng các test-case sử dụng đồ thị nguyên nhân

Kết quả:

1. Đặc tả được chia thành các phần có thể thực hiện được. Điều này là cần thiết vì kỹ thuật này sẽ khó sử dụng khi được sử dụng trên những đặc tả quá lớn
2. Nguyên nhân và kết quả trong đặc tả được nhận biết. Một nguyên nhân là một trạng thái đầu vào nhất định hoặc cũng có thể là một lớp tương đương của các trạng thái đầu vào. Một kết quả là một trạng thái đầu ra hoặc một sự biến đổi hệ thống nào đó. Để nhận biết các nguyên nhân và kết quả bạn phải đọc phần đặc tả, sau đó gạch chân các từ hoặc cụm từ mô tả nguyên nhân và kết quả. Khi đó mỗi nguyên nhân và kết quả này sẽ được gán bởi một số duy nhất.
3. Xây dựng đồ thị nguyên nhân-kết quả bằng cách phát triển và biến đổi nội dung ngữ nghĩa của đặc tả thành đồ thị Boolean nối giữa nguyên nhân và kết quả.
4. Chuyển đồ thị thành một bảng quyết định mục vào giới hạn. Mỗi cột trong bảng mô tả một ca kiểm thử.
5. Các cột trong bảng quyết định được chuyển thành các ca kiểm thử.

Câu 3 : Phương pháp phân lớp tương đương là 1 phương pháp kiểm thử hộp đen dựa trên nguyên tắc chia miền đầu vào của một chương trình thành các lớp dữ liệu, để từ đó lập ra các ca kiểm thử theo mỗi lớp đó. Mục tiêu của phương pháp này là tìm ra 1 ca kiểm thử mà làm lộ ra 1 lớp lỗi. Vì vậy sẽ giảm tổng số các trường hợp kiểm thử phải được xây dựng. Thiết kế ca kiểm thử cho phân lớp tương đương dựa trên sự đánh giá về các lớp tương đương với 1 điều kiện nào đó. Lớp tương đương biểu thị cho tập trạng thái hợp lệ (tập trạng thái mô tả các đầu vào hợp lệ của chương trình) hay không hợp lệ (tập trạng thái mô tả tất cả các trạng thái có thể khác của điều kiện) đối với các điều kiện vào. Chẳng hạn như thông tin về tên của một người Việt, lớp tương đương hợp lệ bao gồm tập các tên riêng tiếng Việt hợp lệ có độ dài từ 1 đến 7 kí tự. Lớp tương đương không hợp lệ là tập các tên riêng không có trong tiếng Việt hoặc được viết bằng số hay kí tự đặc biệt.

Thiết kế trường hợp thử bằng lớp tương đương được tiến hành theo 2 bước:

1. Xác định các lớp tương đương

Các lớp tương đương được xác định bằng cách lấy mỗi trạng thái đầu vào và phân chia nó thành 2 hay nhiều nhóm. Các nhóm này có thể là các lớp tương đương hợp lệ, các lớp tương đương không hợp lệ,... Với 1 đầu vào hay 1 điều kiện bên ngoài đã cho, việc xác định các lớp tương đương hầu như là một quy trình mang tính kinh nghiệm. Để xác định các lớp tương đương có thể áp dụng theo các nguyên tắc sau:

- Nếu 1 trạng thái đầu vào định rõ giới hạn của các giá trị, xác định 1 lớp tương đương hợp lệ và 2 lớp tương đương không hợp lệ.
- Nếu 1 trạng thái đầu vào xác định số giá trị, xác định 1 lớp tương đương hợp lệ và 2 lớp tương đương bất hợp lệ.
- Nếu 1 trạng thái đầu vào chỉ định một tình huống “chắc chắn”, xác định 1 lớp tương đương hợp lệ và 1 lớp tương đương không hợp lệ. Nếu chương trình xử lí các phần tử trong cùng một lớp là khác nhau thì phải chia lớp tương đương đó thành các lớp tương đương nhỏ hơn.

2. Xác định các ca kiểm thử

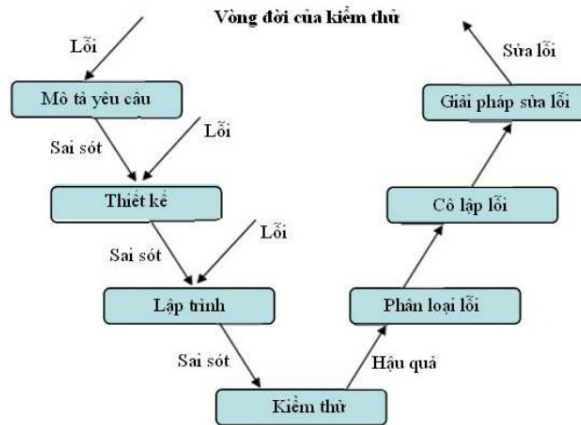
Với các lớp tương đương được xác định ở bước trên, bước tiếp theo là dựa vào các lớp tương đương đó để xác định các ca kiểm thử. Quá trình này được thực hiện như sau:

- Gán 1 số duy nhất cho mỗi lớp tương đương.
- Cho đến khi tất cả các lớp tương đương hợp lệ được bao phủ bởi các ca kiểm thử, viết 1 ca kiểm thử mới bao phủ càng nhiều các lớp tương đương đó chưa được bao phủ càng tốt .
- Cho đến khi các ca kiểm thử của bạn đã bao phủ tất cả các lớp tương đương không hợp lệ, viết 1 ca kiểm thử bao gồm 1 và chỉ 1 trong các lớp tương đương không hợp lệ được bao phủ.

Câu 4: Vòng đời của việc “Kiểm thử phần mềm”. Cho biết hồ sơ của các quá trình kiểm thử tương ứng với các giai đoạn phát triển phần mềm

Lỗi có thể xảy ra trong tất cả các công đoạn từ “Mô tả yêu cầu”, “Thiết kế” đến “Lập trình”. Từ công đoạn này chuyển sang công đoạn khác thường nảy sinh các sai sót (do dư thừa hoặc thiếu theo mô tả yêu cầu).

⇒ Công đoạn kiểm thử chúng ta sẽ phát hiện ra các hậu quả (các kết quả không mong muốn). Quá trình sửa lỗi bao gồm “phân loại lỗi”, “cô lập lỗi” (tìm ra nguyên nhân và nơi gây lỗi), đề ra “giải pháp sửa lỗi” và cuối cùng là khắc phục lỗi



Câu 5: Hãy phân biệt “Thông tin cấu hình phần mềm” và “Thông tin cấu hình kiểm thử” . Chỉ rõ sự khác biệt giữa “Testing” và “Debug”

Thông tin về cấu hình phần mềm: các thông tin này bao gồm: mô tả về yêu cầu của phần mềm (Software Requirement Specification). Mô tả về thiết kế của chương trình (Design Specification) và mã của chương trình. Liên quan đến đặc điểm của phần mềm

Thông tin về cấu hình kiểm thử: bao gồm kế hoạch kiểm thử, và thủ tục kiểm thử và các chương trình chạy kiểm thử như: chương trình giả lập môi trường, chương trình tạo các trường hợp kiểm thử... Các trường hợp kiểm thử phải đi cùng với kết quả mong muốn, Trong thực tế những thông tin này cũng là một phần của Thông tin cấu hình của một phần mềm ở trên. Liên quan đến hoạt động của phần mềm

*Sự khác biệt giữa Testing và Debug

- **Testing** là quá trình kiểm tra xem phần mềm có đáp ứng được các yêu cầu kỹ thuật, chức năng và người dùng hay không, bằng cách thực thi phần mềm với các trường hợp kiểm thử và so sánh kết quả thực tế với kết quả mong đợi. Mục đích của testing là phát hiện ra các lỗi, sai sót hoặc hư hỏng trong phần mềm.






- **Debug** là quá trình sửa chữa các lỗi, sai sót hoặc hư hỏng trong phần mềm. Debugging bắt đầu sau khi testing phát hiện ra các vấn đề trong phần mềm và kết thúc khi phần mềm hoạt động đúng như mong muốn. Debugging yêu cầu phân tích nguyên nhân gốc của các vấn đề và áp dụng các giải pháp để khắc phục

Tóm lại, testing và debug là hai quá trình khác nhau trong phần mềm và kiểm thử. Testing là để tìm ra các vấn đề trong phần mềm, trong khi debugging là để giải quyết các vấn đề đó. Hai quá trình này làm việc cùng nhau để tạo ra một sản phẩm chất lượng cao và đáp ứng nhu cầu của người dùng.

PHƯƠNG PHÁP KIỂM THỬ MODULE

Các đoạn chương trình thường gặp

<pre> graph TD A["{A}"] -.-> Bn["{Bn}"] Bn -.-> Bn1["{Bn-1}"] Bn1 -.-> B1["{B1}"] B1 -.-> B["{B}"] B -.-> Bn </pre>	<p>Trong đó các mệnh đề $\{B_i\}$ được xác định như sau:</p> <p>$\{B_1\}$ là mệnh đề $\{B[x_n E_n]\}$</p> <p>...</p> <p>$\{B_{n-1}\}$ là mệnh đề $\{B_{n-2}[x_2 E_2]\}$</p> <p>$\{B_n\}$ là mệnh đề $\{B_{n-1}[x_1 E_1]\}$</p> <p>Chú ý: Trong trường hợp $\{A\}$ không logic $\{B_n\}$ thì ta nói P là có lỗi.</p>
--	---

 <pre> 1 {A} 2 if E then P; 3 {B} </pre>	$\{A, E\} \vdash \{B\}$ và $\{A, \neg E\} \vdash \{B\}$
 <pre> 1 {A} 2 if E then P; 3 else Q; 4 {B} </pre>	$\{A, E\} \vdash \{B\}$ và $\{A, \neg E\} \vdash \{B\}$
 <pre> 1 {A} 2 while E do P; 3 {B} </pre>	<ul style="list-style-type: none"> • Dự trù: $\{C\}$ • $\{C, E\} \vdash \{C\}$ • Kết thúc có được: $\{C, \neg E\}$
 <pre> 1 {A} 2 do P while E; 3 {B} </pre>	<ul style="list-style-type: none"> • Dự trù: $\{C\}$ • $\{C, E\} \vdash \{C\}$ • Kết thúc có được: $\{C, \neg E\}$
 <pre> 1 {A} 2 For i = F1 to F2 do P; 3 {B} </pre>	<ul style="list-style-type: none"> • Dự trù $\{C\}$ • $\{A\} \Rightarrow i = F1 \vdash \{C\}$ • $t = F2$ • $\{C, i \leq t\} \vdash \{C\}$ • Kết thúc có được: $\{C, i > t\}$

Luyện tập

Câu 1

```

1  {A}
2    P1;
3    if E1 then P1
4    else
5        P2;
6    while E2 do P3;
7    P2;
8    x1 := F1;
9    x2 := F3;
10 {B}

```

Bước 1: Ta có chương trình P như sau:

```

P1;
if E1 then P1 else P2;
while E2 do P3;
P2;
x1:= F1;
x2 := F3;

```

Bước 2: Ta có được đầu vào $\{A\}$ và đầu ra $\{B\}$. Ta cần chứng tỏ $\{A\} P \{B\}$

Bước 3: Dự trữ $\{C\}$ là bất biến của đoạn trình while E2 do P3 trong điều kiện E2

Bước 4: Cần có $\{A\} P1; \text{if } E1 \text{ then } P1 \text{ else } P2; \{C\}$ thì chứng minh:

- $\{A\} P1 \{A'\}$ và $\{A', E1\} P1 \{C\}$
- $\{A\} P1 \{A'\}$ và $\{A', !E1\} P1 \{C\}$

Do đó: $\{A\} P1, \text{if } E1 \text{ then } P1 \text{ else } P2; \{C\}$

Bước 5: Để chứng minh $\{C\}$ bất biến với đoạn trình while E2 do P3

ta cần có $\{C, E2\} P3 \{C\}$

Bước 6: Có $\{C, !E2\}$

Bước 7: Ta có $\{C, !E2\} P2; x1 := F1; x2 := F3; \{B\}$

Theo tính chất của phép gán:

- $\{B[x2|F3]\} = \{B1\}$
- $\{B1[x1|F1]\} = \{B2\}$

Và $\{C, !E2\} P2 \{B2\}$

Rõ ràng ta có: $\{B2\} \Rightarrow_L \{B\}$, nên $\{C, !E2\} P2; x1 := F1; x2 := F3; \{B\}$

Vậy $\{A\} \Rightarrow_L \{B\}$ nên $\{A\} P \{B\}$

Câu 2

```
1  {A}
2    P1;
3    if E1 then P1 else P2;
4    while E2 do {
5        P3;
6        while E3 do P4;
7        if E4 then x := F3;
8    }
9    P2;
10   x1 := F1;
11   x2 := F3;
12  {B}
```

Bước 1: Ta có đoạn chương trình P như sau:

```
{A}
P1;
if E1 then P1 else P2;
while E2 do {
    P3;
    While E3 do P4;
    if E4 then x:= F3;
```

```

    }
    P2;
    x1 := F1;
    x2 = F3;
    {B}

```

Bước 2: Ta có được đầu vào $\{A\}$ và đầu ra $\{B\}$. Cần chứng tỏ $\{A\} \vdash \{B\}$

Bước 3: Dự trù $\{C\}$ bất biến với đoạn trình sau:

```

while E2 do {
    P3;
    While E3 do P4;
    if E4 then x := F3;
}

```

trong điều kiện E2

Bước 4: Cần có $\{A\} \vdash P1$; if E1 then P1 else P2; $\{C\}$ thì chứng minh

- $\{A\} \vdash P1$ $\{A'\}$ và $\{A', E1\} \vdash P1$ $\{C\}$
- $\{A\} \vdash P1$ $\{A'\}$ và $\{A', !E1\} \vdash P2$ $\{C\}$

Do đó: $\{A\} \vdash P1$; if E1 then P1 else P2; $\{C\}$

Bước 5: Để chứng minh $\{C\}$ bất biến với đoạn chương trình sau

```

while E2 do {
    P3;
    while E3 do P4;
    if E4 then x := F3;
}

```

Ta cần có: $\{C, E2\} \vdash P3$; while E3 do P4; if E4 then x := F3; $\{C\}$

Dự trù $\{D\}$ là bất biến của đoạn chương trình while E3 do P4, trong điều kiện E3

Chứng tỏ: $\{D, E3\} \vdash P4$ $\{D\}$

Ta có: $\{C, E2\} \vdash P3$ $\{D\}$

$\{D, E3\} \vdash P4$ $\{D\}$

Có $\{D, !E3\}$ dừng vòng while

- $\{D, !E3, E4\} x := F3 \{C\}$
- $\{D, !E3, !E4\} \Rightarrow L \{C\}$

Do đó: $\{C, E2\} \Rightarrow L \{C\}$

Bước 6: $\{C, !E2\}$

Bước 7: Ta có $\{C, !E2\} P2; x1 := F1; x2 = F3; \{B\}$

Theo tính chất của phép gán:

- $\{B[x2|F3]\} = \{B1\}$
- $\{B1[x1|F1]\} = \{B2\}$

Và ta có $\{C, !E2\} P2 \{B2\}$

Rõ ràng ta có: $\{B2\} \Rightarrow L \{B\}$ nên $\{A\} P2; x1 := F1; x2 = F3; \{B\}$

Vậy $\{A\} \Rightarrow L \{B\}$ nên $\{A\} P \{B\}$

Câu 3

```
1  {A}
2  P1;
3  if E1 then P1
4  else {
5      while E5 do P4;
6      P6;
7  }
8  while E2 do {
9      P3;
10     while E3 do P4;
11     if E4 then x := F3;
12 }
13 P2;
14 while E2 do {
15     x1 := F4;
16     P2;
17 }
18 x1 := F1;
19 x2 := F3;
20 {B}
```

Bước 1: Ta có đoạn chương trình P sau:

```

P1;
if E1 then P1
else {
    while E5 do P4;
    P6;
}
while E2 do {
    P3;
    While E3 do P4;
    if E4 then x := F3;
}
P2;
while E2 do {
    x1 := F4;
    P2;
}
x1 := F1;
x2 := F3;

```

Bước 2: Ta có đầu vào $\{A\}$ và đầu ra $\{B\}$. Ta cần chứng tỏ $\{A\} \vdash \{B\}$

Bước 3: Dự trữ $\{C\}$ là bất biến của đoạn trình:

```

While E2 do {
    P3;
    while E3 do P3;
    if E4 then x := F3;
}

```

Với điều kiện E2

Bước 4: Ta cần chứng minh

$\{A\}$

```

    if E1 then P1
  else {
    while E5 do P4;
    P6;
  }
{C}

```

Dự trữ D là bất biến của đoạn trình: while E5 do P4; với điều kiện E5

Mà ta có Q1 là đoạn trình while E5 do P4; P6

Ta có:

- $\{A, E1\} P1 \{C\}$
- $\{A, !E1\} Q1 \{C\}$ ta cần chứng tỏ $\{A, !E1\} \text{ while } E5 \text{ do } P4; P6; \{C\}$
 - Chứng tỏ: $\{A, !E1, E5\} P4 \{D\}$
 - Có $\{A, !E1, !E5\}$
 - $\{A, !E1, !E5\} P6 \{C\}$ nên $\{A, !E1\} \Rightarrow_L \{C\}$
- Vậy ta có thể nói rằng $\{A, !E1\} Q1 \{C\}$

Để chứng tỏ $\{C\}$ bất biến qua đoạn trình

```

while E2 do {
  P3;
  while E3 do P4;
  if E4 then x := F3;
}

```

Ta cần chứng tỏ: $\{C, E2\} P3; \text{ while } E3 \text{ do } P4; \text{ if } E4 \text{ then } x := F3; \{C\}$

Ta có:

- $\{F\}$ là dự trữ của đoạn chương trình while E3 do P4 với điều kiện E3
- Ta có $\{C, E2\} P3 \{F\}$
- Ta có cần chứng tỏ: $\{F, E3\} P4 \{F\}$
- Có $\{F, !E3\}$

- $\{F, !E3, E4\} x := F3 \{C\}$
- $\{F, !E3, !E4\} \Rightarrow L \{C\}$

Do đó: $\{C, E2\} P3; \text{while } E3 \text{ do } P4; \text{if } E4 \text{ then } x := F3; \{C\}$

Bước 5: Có $\{C, !E2\}$

Bước 6: Dự trù $\{G\}$ là bất biến của đoạn chương trình sau

```
while E2 do {
    x1 := F4;
    P2;
}
```

Với điều kiện E2

Bước 7: Để chứng minh $\{G\}$ là bất biến của đoạn chương trình while E2 do $\{x1 := F4; P2\}$

Ta cần có: $\{C, !E2\} P2 \{G\}$

Ta có chứng minh $\{G, E2\} x1 := F4; P2; \{G\}$

- $\{G\} P2 \{G'\}$ và $\{G'[x|F4]\} = \{G1\}$

Mà $\{G1\} \Rightarrow L \{G\}$ nên $\{G, E2\} \Rightarrow L \{G\}$

Do đó $\{G, E2\} x := F4; P2 \{G\}$

Bước 7: Có $\{G, !E\}$

Bước 8: Ta chứng tỏ $\{G, !E\} x1: F1, x2:=F3 \{B\}$

Theo tính chất của phép gán:

- $\{B[x1|F1]\} = \{B1\}$
- $\{B1[x2|F3]\} = \{B2\}$

Do đó $\{B2\} \Rightarrow L \{B\}$ nên $\{G, !E\} \Rightarrow L \{B\}$

Vậy $\{A\} \Rightarrow L \{B\}$ nên $\{A\} P \{B\}$

Câu 4:


```

1  {A}
2      if E1 then P1;
3      P2;
4      P3;
5      while E2 do {
6          if E3 then P3
7          else P4;
8          while E4 do {
9              P5;
10             x = F1;
11         }
12     }
13     if E1 then P1
14     else P2;
15     while E3 do {
16         y = F2;
17         if E4 then P5
18         else P3;
19     }
20 {B}

```

Bước 1: Ta có đoạn trình P như sau:

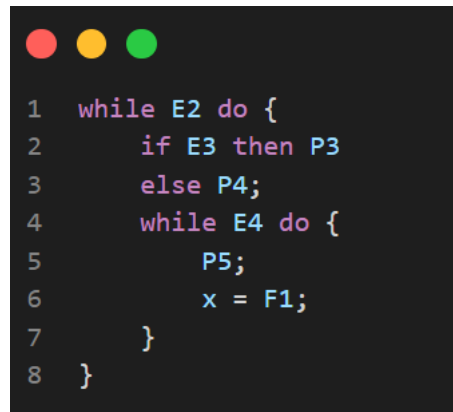
```

1  if E1 then P1;
2  P2;
3  P3;
4  while E2 do {
5      if E3 then P3
6      else P4;
7      while E4 do {
8          P5;
9          x = F1;
10     }
11 }
12 if E1 then P1
13 else P2;
14 while E3 do {
15     y = F2;
16     if E4 then P5
17     else P3;
18 }
19

```

Bước 2: Ta có được đầu vào $\{A\}$ và đầu ra $\{B\}$. Ta cần chứng tỏ $\{A\} \vdash \{B\}$

Bước 3: Dự trù $\{C\}$ là bất biến của đoạn chương trình sau:



```
1  while E2 do {
2      if E3 then P3
3      else P4;
4      while E4 do {
5          P5;
6          x = F1;
7      }
8  }
```

với điều kiện là E2

Bước 4: Ta cần có $\{A\} \vdash \text{if } E2 \text{ then } P1; P2; P3; \{C\}$. Chứng minh

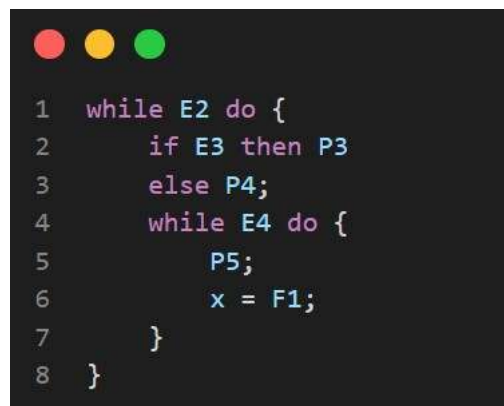
Ta có:

- $\{A, E2\} \vdash P1 \vdash \{A'\}$
- $\{A, !E2\} \Rightarrow_L \{A'\}$

Ta chứng tỏ: $\{A'\} \vdash P1, P2 \vdash \{C\}$

Nên: $\{A\} \vdash \text{if } E2 \text{ then } P1; P2; P3 \vdash \{C\}$

Bước 5: Chứng minh $\{C\}$ bất biến với đoạn trình:



```
1  while E2 do {
2      if E3 then P3
3      else P4;
4      while E4 do {
5          P5;
6          x = F1;
7      }
8  }
```

Ta có: $\{C, E2\} \vdash \text{if } E3 \text{ then } P3 \text{ else } P4; \text{ while } E4 \text{ do } \{P5; x = F1;\} \vdash \{C\}$

Dự trù $\{D\}$ bất biến với đoạn trình:

```

1  while E4 do {
2      P5;
3      x = F1;
4  }

```

Ta cần có:

- $\{C, E2, E3\} P3 \{D\}$
- $\{C, E2, !E3\} P4 \{D\}$

Do đó: $\{C, E2\} \Rightarrow L \{D\}$ nên $\{C, E2\} \text{ if } E3 \text{ then } P3 \text{ else } P4; \{D\}$

Chúng minh $\{D\}$ bất biến while $E4 \text{ do } \{P5; x = F1;\}$ với điều $E4$. Ta cần chứng tỏ khi: $\{D, E4\} P5; x = F1; \{D\}$

Theo tính chất của phép gán:

- $\{D[x|F1]\} = \{D1\}$
- $\{D, E4\} P5 \{D1\}$

Do đó: $\{D, E4\} \Rightarrow L \{D\}$ nên $\{D, E4\} P5; x = F1; \{D\}$

Có được: $\{D, !E4\}$, kết thúc while với điều $E4$ không thỏa

Ta cần chứng tỏ: $\{C, E2, D, !E4\} \Rightarrow L \{C\}$

Do đó: $\{C, E2\} \Rightarrow L \{C\}$

Nên: $\{C, E2\} \text{ if } E3 \text{ then } P3 \text{ else } P4; \text{ while } E4 \text{ do } \{P5; x = F1;\} \{C\}$

Bước 6: Có được $\{C, !E2\}$

Bước 7: Dự trù G là bất biến của đoạn chương trình sau:

```

1  while E3 do {
2      y = F2;
3      if E4 then P5
4      else P3;
5  }

```

Với điều kiện $E3$

Bước 8: Ta cần có $\{C, !E2\} \text{ if } E1 \text{ then } P1 \text{ else } P2; \{G\}$

Ta có:

- $\{C, !E2, E1\} P1 \{G\}$
- $\{C, !E2, !E1\} P2 \{G\}$

Do đó: $\{C, !E2\} \Rightarrow L \{G\}$ nên $\{C, !E2\}$ if E1 then P1 else P2; $\{G\}$

Bước 9: Ta chứng minh $\{G, E3\} y = F2; \text{if } E4 \text{ then } P5 \text{ else } P3; \{G\}$

Ta có:

- $\{G, E3\} y = F2; \{G1, E3\}$

Theo tính chất của phép gán

$\{G1[y|F2], E3\} = \{G, E3\}$

$\{G1[y|F2], E3\}$ if E4 then P5 else P3; $\{G\}$

- $\{G1, E3, E4\} P5 \{G\}$
- $\{G1, E3, !E4\} P3 \{G\}$

Do đó: $\{G, E3\} y = F2; \text{if } E4 \text{ then } P5 \text{ else } P3; \{G\}$

Bước 10: Có được $\{G, !E3\}$ dễ dàng ta thấy được $\{G, !E3\} \Rightarrow L \{B\}$
nên $\{A\} P \{B\}$

Vậy $\{A\} P \{B\}$ hay đoạn trình trên là đúng

Câu 5

```
1  {A}
2  y := F1;
3  if E1 then {x2 := F2; P1;}
4  while E2 do {
5      if E4 then x2 := F2;
6      P3;
7      while E3 do {
8          P4;
9          x1 := F1;
10     }
11     P4;
12 }
13 x2 := F3;
14 if E1 then P1
15 else {
16     do {
17         P4;
18         y2 := F4;
19     }
20     while E5;
21 }
22 P1;
23 P2;
24 while !E2 do {
25     P2;
26     x1 := F2;
27 }
28 For (i := F7; E3; i++) do {P2; x1 := F2;}
29 x1 := F1;
30 P2;
31 {B}
```

Bước 1: Ta có đoạn chương trình B như sau:

```
1  y := F1;
2  if E1 then {x2 := F2; P1;}
3  while E2 do {
4      if E4 then x2 := F2;
5      P3;
6      while E3 do {
7          P4;
8          x1 := F1;
9      }
10     P4;
11 }
12 x2 := F3;
13 if E1 then P1
14 else {
15     do {
16         P4;
17         y2 := F4;
18     }
19     while E5;
20 }
21 P1;
22 P2;
23 while !E2 do {
24     P2;
25     x1 := F2;
26 }
27 For (i := F7; E3; i++) do {P2; x1 := F2;}
28 x1 := F1;
29 P2;
```

Bước 2: Ta có được đầu vào $\{A\}$ và đầu ra $\{B\}$. Chứng tỏ $\{A\} P \{B\}$

Bước 3: Dự trù $\{C\}$ là bất biến của đoạn chương trình

```
1  while E2 do {
2      if E4 then x2 := F2;
3      P3;
4      while E3 do {
5          P4;
6          x1 := F1;
7      }
8      P4;
9  }
```

Với điều kiện E2

Bước 4: Ta cần có $\{A\} y := F1; \text{if } E1 \text{ then } \{x2 := F2; P1;\} \{C\}$

Ta có: $\{A\} y := F1 \{A'\}$

$\{A'\} \text{ if } E4 \text{ then } \{x2 := F2; P1;\} \{C\}$

- $\{A', E4\} \{x2 := F2; P1\} \{C\}$
- $\{A', !E4\} \Rightarrow L \{C\}$

Ta cần chứng tỏ: $\{A', E4\} \{x2 := F2; P1\} \{C\}$

$\{A', E4\} x2 := F2 \{A'', E4\}$

Theo tính chất của phép gán

- $\{A''[x2:=F2], E4\} = \{A', E4\}$

$\{A', E4\} P1 \{C\}$

Do đó: $\{A', E4\} \{x2 := F2; P1\} \{C\}$

Bước 5: Chứng minh $\{C\}$ bất biến qua đoạn chương trình sau:

```
1  while E2 do {
2      if E4 then x2 := F2;
3      P3;
4      while E3 do {
5          P4;
6          x1 := F1;
7      }
8      P4;
9  }
```

$\{C, E2\} \text{ if } E4 \text{ then } x2 := F2; P3; \text{ while } E3 \text{ do } \{P4; x1 := F1;\} P4; \{C\}$

Dự trù $\{D\}$ là bất biến của đoạn chương trình sau:

```
1  while E3 do {
2      P4;
3      x1 := F1;
4  }
```

Với điều kiện E3

Ta cần có: $\{C, E2\} \text{ if } E4 \text{ then } x2 := F2; P3; \{D\}$

$\{C, E2, E4\} x2 := F2 \{C', E2, E4\}$

Theo tính chất phép gán:

- $\{C'[x2|F2], E2, E4\} = \{C, E2, E4\}$

Ta có: $\{C'[x2|F2], E2, E4\} \vdash \{D\}$

Do đó: $\{C, E2\} \text{ if } E4 \text{ then } x2 := F2; P3; \{D\}$

Chứng minh $\{D\}$ là bất biến qua đoạn chương trình

```
1 P4;  
2 x1 := F1;
```

$\{D, E3\} \vdash P4; x1 := F1 \vdash \{D\}$

Theo tính chất của phép gán:

- $\{D[x1|F1]\} = \{D1\}$

$\{D, E3\} \vdash P4 \vdash \{D1\}$

Do đó: $\{D, E3\} \vdash P4; x1 := F1 \vdash \{D\}$

$\{D, !E3\} \vdash P3 \vdash \{C\}$

Do đó: $\{C, E2\} \text{ if } E4 \text{ then } x2 := F2; P3; \text{ while } E3 \text{ do } \{P4; x1 := F1;\} P4; \{C\}$

Bước 6: Có $\{C, !E2\}$

Bước 7: Dự trù G là bất biến của đoạn chương trình sau:

```
1 while !E2 do {  
2     P2;  
3     x1 := F2;  
4 }
```

Với điều kiện !E2

Bước 8: Cần có $\{C, !E2\}$

```

1  x2 := F3;
2  if E1 then P1
3  else {
4      do {
5          P4;
6          y2 := F4;
7      }
8      while E5;
9  }
10 P1;
11 P2;

```

{G}

{C, !E2} x2 := F3; {C'', !E2}

Theo tính của phép gán:

{C''[x2|F3], !E2} = {C, !E2}

{ C''[x2|F3], !E2, E1} P1 {G1}

{ C''[x2|F3], !E2, E1} do {P4; y2 := F4;} while E5 {G1}

Dự trữ: H là bất biến của đoạn chương trình

```

1  do {
2      P4;
3      y2 := F4;
4  }
5  while E5;

```

Với điều kiện E5

Ta chứng tỏ: H là bất biến

{H, E5} P4; y2 := F4; {H}

{H, E5} P4 {H1}

{H[y2|F4]} = {H1}

Nên: {H, E5} P4; y2 := F4; {H}

Ta có: {H, !E5}

Do đó: {H, !E5} =>L {G1}

$\{G1\} P2 \{G2\}$

$\{G2\} P3 \{G\}$

Bước 9: Chứng G là bất biến qua đoạn trình sau:

```
while !E2 do {  
    P2;  
    x1 := F2;  
}
```

Ta có: $\{G, !E2\} P2; x1 := F2; \{G\}$

$\{G[x|F2]\} = \{G1\}$

Mà: $\{G, !E2\} P2 \{G1\}$

Do đó: $\{G, !E2\} P2; x1 := F2 \{G\}$

Bước 10: Có $\{G, !E2\}$

Bước 11: Dự trù $\{T\}$ là bất biến của đoạn chương trình sau:

```
For (i := F7; E3; i++) do {P2; x1 := F2;}
```

Bước 12: Ta có: $\{G, !E2\} \Rightarrow_L \{T\}$

Bước 13:

$\{G, !E2\} \Rightarrow i = F7 \{T\}$

$t = E3$

$\{T, i \leq t\} P2; x1 := F2 \{T\}$

$\{T[x|F2]\} = \{T1\}$

$\{T, i \leq t\} P2 \{T1\}$

Do đó: $\{T, i \leq t\} P2; x1 := F2 \{T\}$

Bước 14: Có $\{T, i > t\}$

```
x1 := F1;  
P2;
```

Bước 15: $\{T, i > t\} x1 := F1; P2; \{B\}$

Ta chứng tỏ: $\{T, i > t\} x1 := F1 \{T1, i > t\}$

Theo tính chất của phép gán:

$\{T1[x1|F1], i > t\} = \{T, i > t\}$

$\{T1, i > t\} \text{ P2 } \{T2\}$

Do đó: $\{T, i > t\} \text{ x1} := F1; \text{ P2}; \{T2\}$

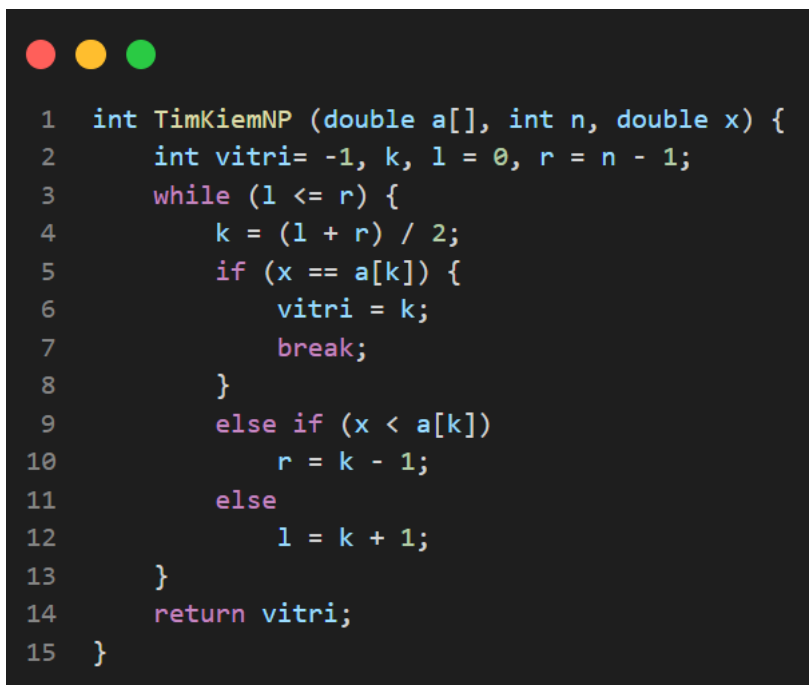
Bước 16: Có được $\{T2\}$ dễ dàng ta thấy được $\{T2\} \Rightarrow L \{B\}$

nên $\{A\} \text{ P } \{B\}$

Vậy $\{A\} \text{ P } \{B\}$ hay đoạn trình trên là đúng

PHƯƠNG PHÁP KIỂM THỬ HỘP TRẮNG

Câu 1: Tìm kiếm nhị phân bằng phương pháp kiểm thử hộp trắng



```
1  int TimKiemNP (double a[], int n, double x) {
2      int vitri= -1, k, l = 0, r = n - 1;
3      while (l <= r) {
4          k = (l + r) / 2;
5          if (x == a[k]) {
6              vitri = k;
7              break;
8          }
9          else if (x < a[k])
10             r = k - 1;
11         else
12             l = k + 1;
13     }
14     return vitri;
15 }
```

Ta thiết kế các trường hợp kiểm thử cho hàm TimKiemNP:

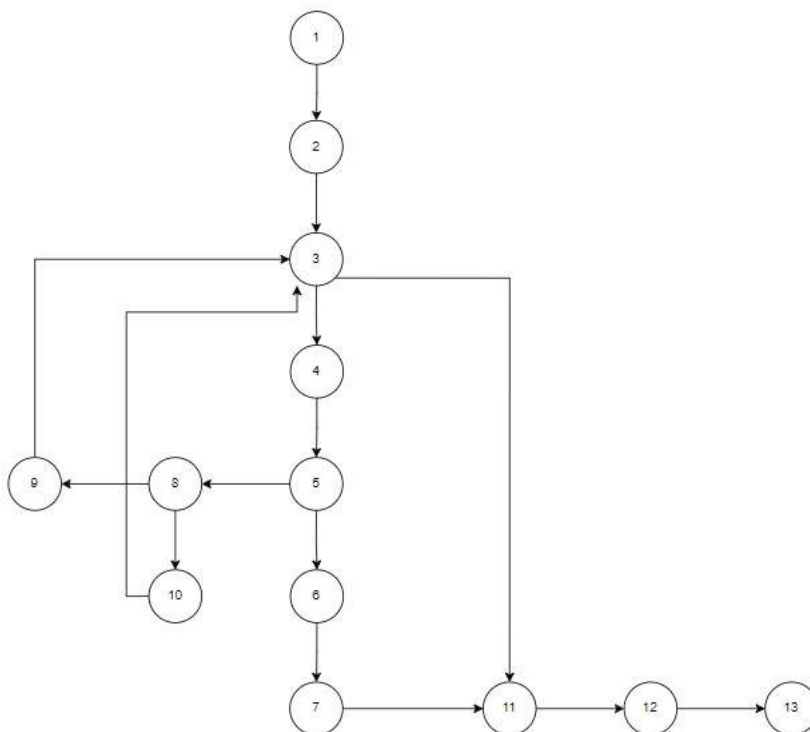
Xác định các nút

```

1  int TimKiemNP (double a[], int n, double x) {
2      int vitri= -1, k, l = 0, r = n - 1;
3      while (l <= r) {
4          k = (l + r) / 2;
5          if (x == a[k]) {
6              vitri = k;
7              break;
8          }
9          else if (x < a[k])
10             r = k - 1;
11         else
12             l = k + 1;
13     }
14     return vitri;
15 }

```

Vẽ đồ thị thể hiện đường diễn tiến của chương trình



Xác định số đường kiểm thử: $V(G)$: $V(G) = \text{Tổng đường} - \text{Tổng nút} + 2$

$$\Rightarrow V(G) = 15 - 13 + 2 = 4$$

Các đường kiểm thử là

1. 2. 3. 11. 12

1. 2. 3. 4. 5. 6. 7. 11. 12

1. 2. 3. 4. 8. 10. 3. 5 ... 3. 11. 12

1. 2. 3. 4. 8. 10. 3. 5. 8 ... 3. 4. 5. 6. 7. 11. 12

Xác định các Test Case:

Đường kiểm thử	Giá trị đầu vào	Kết quả mong đợi
1	$a[] = \{1, 2, 3, 4, 6\}$, $n = -2$, $x = 3$	vitri = -1
2	$a[] = \{1, 2, 3\}$, $n = 3$, $x = 2$	vitri = 1
3	$a[] = \{1, 2, 3\}$, $n = 3$, $x = 7$	vitri = -1
4	$a[] = \{1, 2, 3\}$, $n = 3$, $x = 3$	vitri = 1