	<b>VIETTEL AI RACE</b>	TD062
	<b>KIỂM THỬ GIÁ TRỊ BIÊN</b>	Lần ban hành: 1

## 1. Kiểm thử giá trị biên

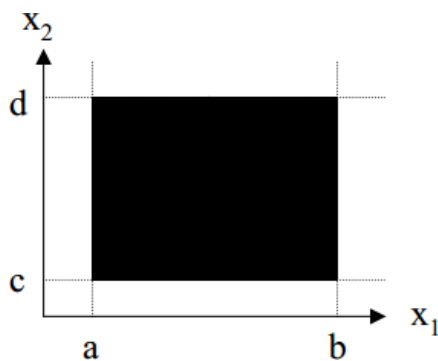
Kiểm thử giá trị biên (boundary value testing) là một trong những kỹ thuật được áp dụng phổ biến nhất trong cách tiếp cận kiểm thử chức năng (kiểm thử hộp đen). Trong thực tế, các lỗi hay xảy ra ở các giá trị biên hoặc cận biên của các biến đầu vào của chương trình cần kiểm thử. Kỹ thuật kiểm thử giá trị biên được đề xuất nhằm phát hiện những lỗi này. Chúng ta sẽ coi một chương trình là một hàm toán học với đầu vào của chương trình tương ứng với các tham số của hàm và đầu ra của chương trình là giá trị trả về của hàm. Vì hàm toán học là ánh xạ từ miền xác định của hàm đến miền giá trị của hàm, chúng ta sẽ tập trung vào các giá trị biên và cận biên của hai miền đầu vào và đầu ra này của hàm để xây dựng các ca kiểm thử. Khi chúng ta dùng biên đầu ra tức là chúng ta cho các kết quả mong đợi nằm ở trên biên và cận biên đầu ra.

### 1.1 Giá trị biên

Giả sử  $y = f(x_1, x_2)$  với  $x_1, x_2, y \in \mathbb{N}$  là một hàm toán học của một chương trình. Khi đó thông thường  $x_1$  và  $x_2$  có miền xác định thể hiện bằng các biên. Ví dụ:


$$a \leq x_1 \leq b \text{ và } c \leq x_2 \leq d$$

trong đó  $a, b, c, d$  là các hằng số nào đó. Phân tô đậm trong hình 5.2 thể hiện miền xác định của hai biến này.



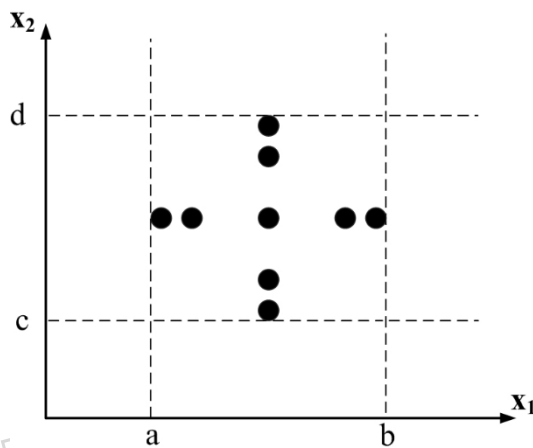
**Hình 5.2: Miền xác định của hàm hai biến.**

Ý tưởng kiểm thử giá trị biên xuất phát từ quan sát nhiều lỗi xảy ra với các giá trị biên này, khi chúng ta không kiểm tra khoảng giá trị hợp lệ của dữ liệu được nhập vào, hay do lỗi lập trình hoặc đặc tả làm các biểu thức điều kiện không chính xác. Ví dụ, đúng ra phải là dấu  $\leq$  nhưng người lập trình hoặc đặc tả lại chỉ viết  $<$  hoặc ngược lại. Kiểm thử với các giá trị biên giúp chúng ta phát hiện các lỗi này. Trong nhiều trường hợp các biên này là ẩn, không được viết rõ ra trong yêu cầu nên lập trình viên dễ mắc lỗi không kiểm tra các giá trị đầu vào, hoặc không kiểm tra kết quả có hợp lệ không trước khi trả về.

	<b>VIETTEL AI RACE</b>	TD062
	<b>KIỂM THỬ GIÁ TRỊ BIÊN</b>	Lần ban hành: 1

Để tăng khả năng phát hiện lỗi, kiểm thử giá trị biên thường lấy năm ca kiểm thử cho mỗi biến là các giá trị: cực đại, cực tiểu, các giá trị cạnh chúng trong miền xác định (gọi là cận biên hoặc cạnh biên), và một giá trị ở giữa miền xác định đại diện cho giá trị thông thường. Chúng ta đánh chỉ số cho chúng lần lượt là  $max$ ,  $min$ ,  $min+$ ,  $max-$  và  $nom$ . Hình 5.3 minh họa các giá trị này.

Thông thường lỗi chương trình xảy ra ngay khi có một sai sót trong phần mềm, chứ không cần kết hợp của nhiều sai sót mới gây ra lỗi. Khi đó các ca kiểm thử theo phương pháp kiểm thử giá trị biên được xây dựng bằng cách lấy một bộ giá trị  $nom$  của các biến, rồi lần lượt thay mỗi giá trị đó của từng biến bằng giá trị biên và



**Hình 5.3: Các ca kiểm thử phân tích giá trị biên cho một hàm hai biến.**


cận biên để tạo ra ca kiểm thử mới. Ví dụ với hàm  $f$  trên ta có bộ kiểm thử sau, thể hiện trên Hình 5.3.

$\{(x1nom, x2nom),$   
 $(x1min, x2nom), (x1min+, x2nom), (x1max-, x2nom), (x1max, x2nom), (x1nom, x2min),$   
 $(x1nom, x2min+), (x1nom, x2max-), (x1nom, x2max)\}$

Tổng quát hóa kiểm thử giá trị biên cho hàm  $n$  biến số và mỗi biến có các giá trị biên và cận biên khác nhau ta có thể dễ thấy sẽ có  $1 + 4n$  ca kiểm thử vì xuất phát từ một ca kiểm thử gồm các giá trị trung bình của các biến, ta thay nó bằng bốn giá trị biên và cận biên:  $min$ ,  $min+$ ,  $max$ , và  $max-$ . Với  $n$  biến sẽ tạo thêm  $4n$  bộ kiểm thử. Do đó tổng số ca kiểm thử là  $1 + 4n$ .

Tuy nhiên tùy theo miền xác định của biến mà số lượng này thực tế có thể ít hơn. Ví dụ biến nguyên thuộc khoảng  $[1, 2]$  thì không có cận biên và không có giá trị ở giữa. Với cách tạo bộ kiểm thử giá trị biên này, mỗi giá trị biên và cận biên xuất hiện một lần với các giá trị trung bình của các biến còn lại, chứ không phải tổ hợp các bộ giá

trị của các biên. Tổ hợp của các biên được gọi là kiểm thử giá trị biên mạnh sẽ được thảo luận trong phần tiếp theo.

	VIETTEL AI RACE	TD062
	KIỂM THỬ GIÁ TRỊ BIÊN	Lần ban hành: 1

Trong thực tế, miền xác định của các biến không chỉ là các miền số, nên chúng ta cần có các vận dụng thích hợp để xác định các giá trị biên tinh tế hơn. Ví dụ trong chương trình NextDate biến tháng month có thể là số nguyên trong khoảng từ 1 đến 12. Nhưng không phải ngôn ngữ lập trình nào cũng cho phép khai báo miền xác định của biến trong khoảng này. Chúng ta cũng có thể dùng kiểu liệt kê (enumeration) để khai báo các tháng bằng tên Januray, ..., December và chúng ta vẫn có giá trị biên và cận biên. Tuy nhiên như trong ngôn ngữ lập trình Java chúng ta phải dùng biến kiểu nguyên int và không khai báo miền giá trị cho chúng được, thì chúng ta sẽ phải dùng các cận của kiểu làm biên. Trong Java, biên của số nguyên là Interger.MIN\_VALUE và Interger.MAX\_VALUE. Trong ngôn ngữ lập trình C, chúng ta cũng có các khoảng giá trị này.

Với biến kiểu Boolean, các giá trị của miền chỉ có TRUE và FALSE, chúng ta không có cận biên và giá trị ở giữa. Với kiểu xâu ký tự, mảng, hay danh sách, tập hợp (collection), chúng ta có thể dựa vào chiều dài/kích thước để làm các giá trị biên. Với các kiểu cấu trúc dữ liệu khác chúng ta có thể xây dựng các giá trị biên và cận biên dựa theo các thành phần của cấu trúc.

**Ưu nhược điểm:** Khi các biến của hàm là độc lập, không có quan hệ ràng buộc lẫn nhau, thì kiểm thử giá trị biên tỏ ra hiệu quả. Nhưng khi chúng có quan hệ phụ thuộc nào đó thì phương pháp này dễ tạo ra các ca kiểm thử không hợp lý, vì các giá trị cực đại, cực tiểu có thể không kết hợp với nhau để tạo thành ca kiểm thử hợp lệ.

Ví dụ, khi  $x_1$  và  $x_2$  của hàm  $f$  trên có các ràng buộc là  $x_1 \in [1, 100]$ ,  $x_2 \in [1, 100]$  and  $x_1 + x_2 \leq 120$  thì một biến đạt cực đại sẽ không thể kết hợp với một biến khác ở giá trị trung bình. Nếu chúng ta xuất phát

từ ca kiểm thử của hai giá trị trung bình (50, 50) thì khi thay giá trị đầu bằng giá trị cực đại sẽ tạo ra ca kiểm thử (100, 50). Ca kiểm thử này không thỏa mãn quan hệ  $x_1 + x_2 \leq 120$ . Ví dụ khác với hàm NextDate thì biến ngày, tháng và năm có ràng buộc. Một số tháng có 31 ngày, một số lại có 30 ngày. Năm nhuận lại có ràng buộc với số ngày trong tháng hai. Nếu không xét sự phụ thuộc này kiểm thử giá trị biên không tạo ra được ca kiểm thử ngày 28 tháng 2 năm 2012, vì 28 không là biên và cận biên của khoảng ngày từ 1 đến 31.


Kiểm thử giá trị biên cũng không quan tâm đến tính chất, đặc trưng của hàm, đồng thời cũng không xét đến ngữ nghĩa của biến hay quan hệ giữa các biến. Nó chỉ máy móc lấy các giá trị biên, cận biên và trung bình để tổ hợp tạo ra các ca kiểm thử. Nên khi áp dụng chúng ta cần xác định sơ bộ các tính chất trên của hàm có không.

Ưu điểm của phương pháp này là đơn giản và có thể tự động hóa việc sinh các ca kiểm thử khi chúng ta đã xác định được các biên của các biến. Với nhiều kiểu dữ liệu cơ bản có sẵn chúng ta cũng có thể xây dựng sẵn các biên này để sử dụng.

## 1.2 Một số dạng kiểm thử giá trị biên

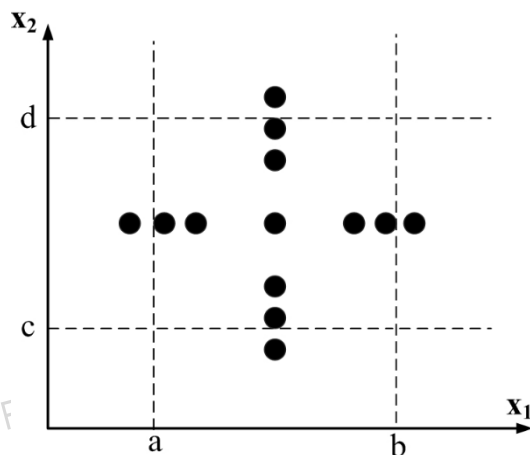
### 1.2.1 Kiểm thử giá trị biên mạnh

Kiểm thử giá trị biên mạnh là mở rộng của kiểm thử giá trị biên bằng việc bổ sung các giá trị cận biên bên ngoài miền xác định. Ngoài 5 giá trị biên đã nêu ở phần trước chúng ta lấy thêm các giá trị cận biên ở ngoài miền xác định là  $max+$  và  $min-$  như trong Hình 5.4. Các ca kiểm

	<b>VIETTEL AI RACE</b>	TD062
	<b>KIỂM THỬ GIÁ TRỊ BIÊN</b>	Lần ban hành: 1

thử này sẽ giúp ta kiểm tra chương trình với dữ liệu không hợp lệ, nằm ngoài khoảng mong đợi. Ví dụ khi nhập ngày 32/1/2013 chương trình cần có thông báo lỗi thích hợp. Các ca kiểm thử mạnh này giúp chúng ta kiểm tra xem chương trình có xử lý các ngoại lệ hay kiểm tra biến đầu vào hay không.

Với các ngôn ngữ lập trình không kiểm tra kiểu khi biên dịch hay dữ liệu được nhập vào bên ngoài, do người sử dụng đưa vào hoặc lấy từ hệ thống khác, việc này là rất cần thiết. Rất nhiều lập trình viên mới vào nghề thường xuyên không kiểm tra dữ liệu đầu vào nên chương trình chỉ chạy với dữ liệu nhập đúng như suy nghĩ của người lập trình. Hậu quả của việc này không chỉ đơn thuần là chương trình bị sai, mà thường gây vi phạm bộ nhớ, dẫn đến tắt chương trình (crash), và là một trong những lỗ hổng an ninh dễ bị khai thác.

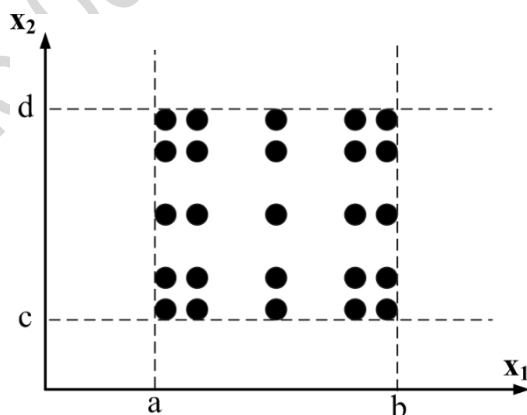


Hình 5.4: Các ca kiểm thử mạnh cho hàm hai biến.

### 1.2.2 Kiểm thử giá trị biên tổ hợp

Ở trên, chúng ta chỉ tạo các ca kiểm thử với giá trị biên và cận biên cho từng biến. Nếu chúng ta mở rộng với hai hoặc với tất cả các biến đều được đẩy đến giá trị biên và cận biên thì chúng ta sẽ tạo ra được các ca kiểm thử giá trị biên tổ hợp. Tức là từ bộ giá trị 5 phần tử  $min$ ,  $min+$ ,  $nom$ ,  $max-$  và  $max$  của mỗi biến ta lấy tích Đề-các (Cartesian) của chúng để tạo ra các ca kiểm thử. Hình 5.5 minh họa các ca kiểm thử giá trị biên tổ hợp của hàm hai biến.

	VIETTEL AI RACE	TD062
	KIỂM THỬ GIÁ TRỊ BIÊN	Lần ban hành: 1



**Hình 5.5: Các ca kiểm thử biên tổ hợp của hàm hai biến.**

Có thể thấy cách tổ hợp các biên và cận biên này sẽ kiểm tra kỹ hơn kiểm thử giá trị biên thông thường. Tuy nhiên số ca kiểm thử theo cách này tăng lên đáng kể, lên đến  $5^n$  ca kiểm thử so với  $4n + 1$  ca theo kiểm thử biên thông thường.

Tương tự như kiểm thử giá trị biên mạnh, ta có thể mở rộng kiểm thử biên tổ hợp với bộ 7 giá trị của kiểm thử giá trị biên mạnh. Chúng ta sẽ kiểm tra được kỹ hơn nhưng cũng mất nhiều công sức hơn, đến  $7^n$  ca kiểm thử.

### 1.2.3 Kiểm thử các giá trị đặc biệt

Kiểm thử các giá trị đặc biệt cũng là một phương pháp phổ biến. Đây cũng là phương pháp trực quan nhất và không theo một khuôn dạng cụ thể nào. Dựa trên hiểu biết về bài toán và miền ứng dụng kết hợp với kinh nghiệm cá nhân, người kiểm thử đưa ra các giá trị kiểm thử. Do đó không có hướng dẫn cụ thể nào cho phương pháp này. Mức độ hiệu quả của phương pháp này phụ thuộc nhiều vào khả năng của người kiểm thử. Trên thực tế các đơn vị phát triển phần mềm vẫn áp dụng phương pháp này, vì nhiều khi nó giúp phát hiện lỗi nhanh, không tốn nhiều công sức.