

## Kiểm thử chức năng

Sinh viên : Nguyễn Tiến Hoàng - 21020123

---

### 1 Mô tả bài toán

Bài toán được em triển khai trong báo cáo lần này mang chủ đề Sử dụng Giải thuật Di truyền ( Genetic Algorithms ) trong việc Tối ưu hóa dự đoán giá cổ phiếu trong Thị trường chứng khoán .

#### 1.1 Tổng quan

Chủ thể trong bài toán cố gắng dự đoán giá sắp tới của cổ phiếu sao cho gần với giá thực tế nhất , thông qua việc "hỏi những hàng xóm xung quanh" và đưa ra quyết định mua hay bán để tối ưu hóa lợi nhuận nhất có thể .

Ta xem xét giá cổ phiếu ban đầu là  $P = 1$  và tăng hay giảm bao nhiêu chỉ phụ thuộc vào số lượng người mua và người bán , được tính theo công thức của thị trường chứng khoán :

$$bt = \frac{\text{buyers agents} - \text{sellers agents}}{\text{Total amount of agents}}$$

Formula 2.: bt variable

$$1 + \frac{e^{bt} - e^{-bt}}{e^{bt} + e^{-bt}}$$

Formula 3.: Correction coeficient

Chủ thể chỉ có thể hỏi một vài người , và những người đó đã đi hỏi những người khác và có những câu trả lời , tạo thành một cộng đồng nhỏ trong quần thể bài toán .

Từ lượng thông tin đã có từ cộng đồng nhỏ , có thể dự đoán được giá của cổ phiếu .

Ví dụ : Quần thể bài toán xem xét với 40 người , 21 người mua và 19 người bán , từ công thức trên ta có thể tính được Giá thực tế của cổ phiếu sẽ tăng lên 1.05

Trong cộng đồng nhỏ của chủ thể có 10 người , 3 người mua và 7 người bán , Giá dự đoán của cổ phiếu sẽ giảm xuống 0.62 , lệch đến 0.43 so với Giá thực tế .

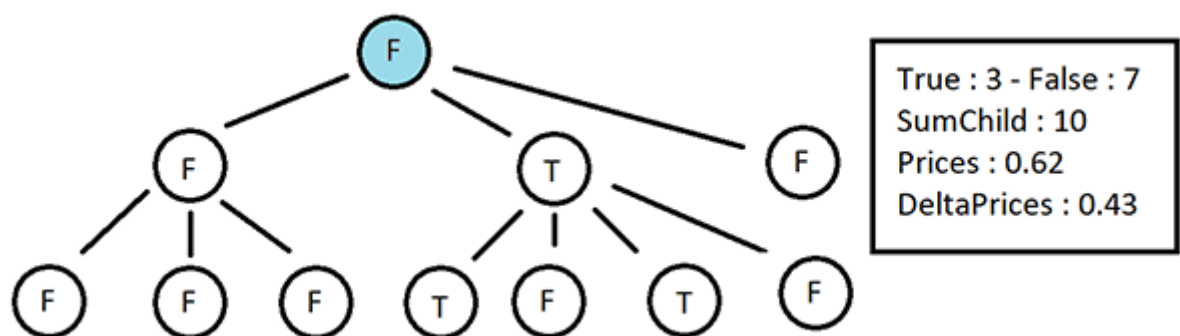
Mục tiêu của bài toán là làm cho độ lệch này nhỏ nhất có thể phụ thuộc vào chủ thể sẽ hỏi những ai .

Độ lệch này thể hiện khả năng đoán đúng và độ uy tín của cộng đồng . Khi chủ thể hỏi những người uy tín , ở trong những cộng đồng chứng khoán uy tín , tỉ lệ thành công sẽ cao hơn rất nhiều so với các trường hợp còn lại .

## 1.2 Khởi tạo quần thể ban đầu

Quần thể khởi tạo bao gồm 40 người , chia làm 4 cộng đồng nhỏ , mỗi cộng đồng có 10 người , tương ứng với 4 trường hợp chủ thể bài toán ở trong cộng đồng nào , đi hỏi những ai .

Mỗi cộng đồng được biểu diễn theo mô hình cây ( Tree ) , thể hiện thông tin sau việc đi hỏi và đưa ra quyết định như hình dưới đây .



Quyết định được đưa ra dựa vào số người trả lời là mua ( true ) lớn hơn hay bán ( false ) lớn hơn .

### 1.3 Chọn lọc tự nhiên

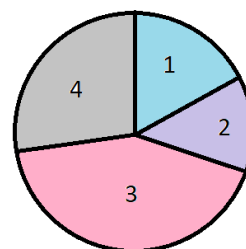
Những cộng đồng sau khi khởi tạo , trả về những thông tin như sau :

Tree	TrueMinusFalse	SumChild	Prices
0	-4.0	10.0	0.6200510377447751
1	6.0	10.0	1.537049566998035
2	2.0	10.0	1.197375320224904
3	-2.0	10.0	0.802624679775096
DeltaPrices	1/DeltaPrices	Percent	
0.4299073372131049	2.326082654189036	15.29745112792527 ( % )	
0.48709119204015505	2.053003659974951	13.501550814353067 ( % )	
0.14741694526702398	6.7834806791624125	44.611468977610606 ( % )	
0.24733369518278403	4.043120769537616	26.589529080111056 ( % )	
GlobalPrices : 1.04995837495788			
Sum : 15.205687762864015			

Việc chọn lọc tự nhiên sẽ được thực hiện dựa trên kết quả chênh lệch giá .

Giá chênh lệch càng ít thì cá thể càng tinh túy và cần được giữ lại để lai ghép .

Cá thể tinh túy sẽ chiếm nhiều tỉ lệ % hơn trong vòng quay chọn lọc và đồng nghĩa với xác xuất được giữ lại là cao hơn .



Ta mong muốn độ chênh lệch giá ( DeltaPrices ) càng nhỏ , và bằng 0 là tối ưu , nên ( 1 / DeltaPrices ) càng lớn sẽ càng tốt . Cá thể nào có ( 1 / DeltaPrices ) càng lớn thì càng chiếm nhiều tỉ lệ % trong vòng quay .

Sau khi đã hoàn thành Chọn lọc tự nhiên , ta "tái tạo" lại quần thể ban đầu gồm 4 cá thể tinh túy nhất .

## 1.4 Lai ghép

Tiến hành lai ghép các cặp cá thể ( 1 và 2 ) , ( 3 và 4 ) để tạo thành 4 cá thể mới mang gen của cả bố và mẹ

Đồng thời hi vọng rằng quần thể con sẽ xuất sắc hơn quần thể trước khi lai tạo .

Mô hình được triển khai bằng cách ngắt cây con trái của cây này gắn vào bên phải của cây kia .

## 1.5 Nghiệm thu

Cuối cùng là tính lại thông số và tiếp tục lặp lại việc Chọn lọc - Lai ghép cho đến khi chênh lệch không quá 3% so với Giá thực tế hoặc các cộng đồng giống hết nhau thì dừng lại .

Kết quả thu được là một cộng đồng uy tín , cho ra kết quả Giá dự đoán gần sát nhất so với Giá thực tế .

Nếu chủ thể hỏi đúng những hàng xóm này , họ sẽ có chiến lược và quyết định đúng đắn để đạt lợi nhuận tối ưu nhất .

## 2 Khảo sát Mã nguồn

### 2.1 Chuẩn mã nguồn trước khi xem xét & phản biện

Tác giả cần tự chuẩn hóa mã nguồn ở mức nhất định , đảm bảo mã nguồn đáp ứng được những yêu cầu cơ bản theo nguyên tắc .

VD : Thụt đầu dòng , Cách đặt tên , Chú thích , .....

Đồng thời , đảm bảo tính nhất quán, dễ hiểu của mã nguồn do nhiều thành viên cùng viết, sửa

Đáp ứng một vài chuẩn / yêu cầu quốc tế cho mã nguồn có thể kể đến như :

- Java / JavaScript Google Style Guide
- BEM HTML DOM Standard
- Refactoring
- Object-oriented
- Clean Code

### 2.2 Phản biện hình thức

Là hình xuất phát cơ bản nhất khi một mã nguồn được đưa ra khảo sát .

Bao gồm những nguyên tắc và hoạt động cơ bản như :

- Xác định vấn đề
- Đặt ra nguyên tắc làm việc
- Chuẩn bị cuộc họp
- Biên bản, báo cáo kết luận

### 2.3 Phản biện chéo ( Phản biện phi hình thức )

Bắt đầu đi sâu vào mã nguồn cụ thể , đơn giản hóa những hình thức cơ bản trong quá trình khảo sát.

Các bên liên quan tham gia khảo sát lẫn nhau và cho ý kiến.

Cách thức hoạt động :

- Tác giả trình bày cho một nhóm nhỏ (5 lập trình, kiểm thử viên khác) đã xem mã nguồn trước
- Thanh tra phản biện mã nguồn
- Đội ngũ Khảo sát viên tham gia cho ý kiến , phản biện qua lại với tác giả và thanh tra phản biện để chắc chắn mã nguồn sẽ đạt được những yêu cầu cần đáp ứng trong đặc tả

## 3 Quy trình khảo sát

### 3.1 Tổng quan

Việc một sản phẩm phần mềm cần phải trải qua một giai đoạn kiểm tra và sửa chữa ở mỗi mốc quan trọng trong vòng đời đường như đã trở thành quy luật tất yếu khi phát triển phần mềm .

Hoàn thành việc viết mã nguồn là một cột mốc quan trọng trong quá trình phát triển nhưng không đồng nghĩa với việc sản phẩm đã hoàn thiện .

Cho dù đã triển khai đầy đủ các đơn vị , thành phần cũng không đảm bảo sản phẩm đạt yêu cầu .

Cột mốc tiếp theo là kiểm tra , hiệu chỉnh tất cả hoặc một số lượng đáng kể các đơn vị tạo thành các thành phần chính của sản phẩm .

Ý tưởng đằng sau việc xem xét là tìm ra các lỗi càng gần điểm xuất phát của chúng càng tốt để các lỗi đó được loại bỏ với ít nỗ lực hơn và sản phẩm tạm thời chứa ít khuyết tật hơn trước khi thực hiện nhiệm vụ tiếp theo.

### 3.2 Định nghĩa

Các đoạn mã được xem xét bằng cách áp dụng các kỹ thuật thường được gọi là kiểm tra và hướng dẫn.

Định nghĩa ban đầu về kiểm tra được đặt ra bởi Michael Fagan và định nghĩa về hướng dẫn của Edward Yourdon :

- Kiểm tra : là một nhóm đồng nghiệp xem xét từng bước một sản phẩm làm việc, với mỗi bước được kiểm tra dựa trên các tiêu chí đã định trước.
- Hướng dẫn: là một bài đánh giá trong đó tác giả dẫn dắt nhóm thực hiện sản phẩm theo cách thủ công hoặc mô phỏng bằng cách sử dụng các tình huống được xác định trước.

Đó là một cách tiếp cận có hệ thống để kiểm tra mã nguồn một cách chi tiết , đánh giá chất lượng của phần mềm bên cạnh rất nhiều cách tiếp cận khác .

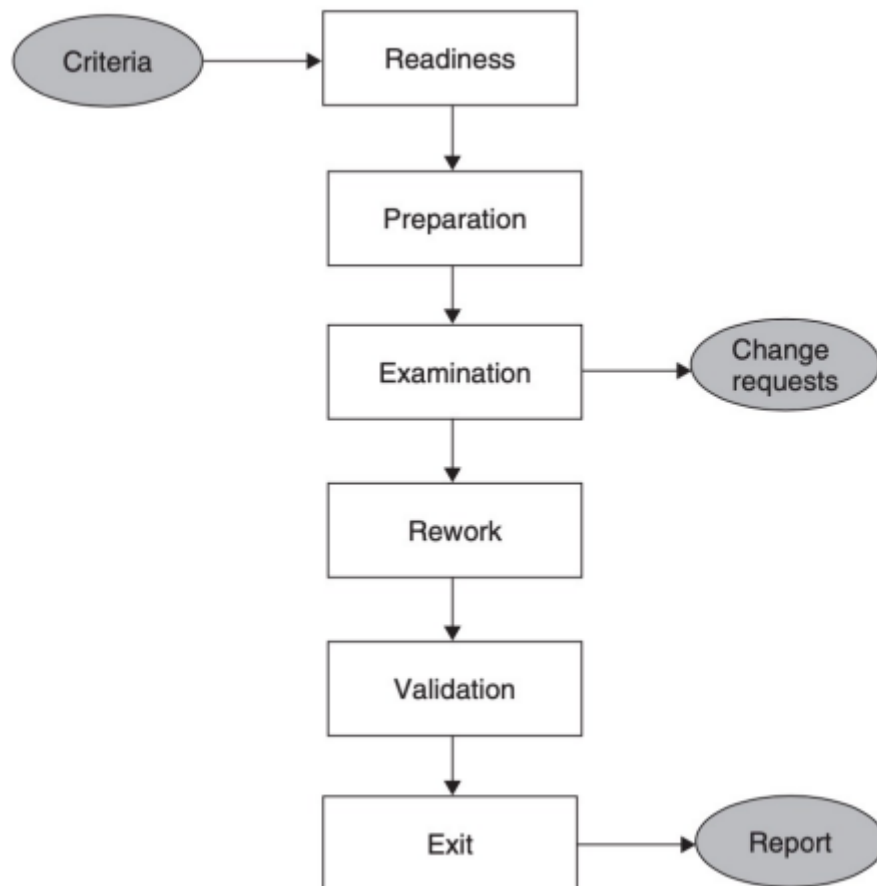
Chìa khóa thành công của việc xem xét mã nguồn là phân chia và chinh phục, nghĩa là, có người kiểm tra kiểm tra các phần nhỏ của đơn vị một cách riêng biệt, đồng thời đảm bảo những điều sau :

- Không có gì bị bỏ qua
- Tính đúng đắn của tất cả các phần được kiểm tra của mô-đun để có thể đảm bảo tính đúng đắn của toàn bộ mô-đun.

### 3.3 Quy trình khảo sát mã nguồn

Các hướng dẫn chung để thực hiện xem xét mã bao gồm sáu bước : sẵn sàng, chuẩn bị, kiểm tra, làm lại, xác nhận và thoát .

Đầu vào cho bước sẵn sàng là các tiêu chí phải được thỏa mãn trước khi bắt đầu quá trình xem xét mã và quá trình này tạo ra hai loại tài liệu, một yêu cầu thay đổi (CR) và một báo cáo.



#### 3.3.1 Sẵn sàng ( Readiness )

Tác giả của đơn vị đảm bảo rằng đơn vị được kiểm tra đã sẵn sàng để xem xét. Một đơn vị được cho là sẵn sàng nếu nó đáp ứng các tiêu chí sau :

- **Tính đầy đủ:** Tất cả các mã liên quan đến thiết bị được xem xét phải có sẵn để những người đánh giá có thể đọc mã và cố gắng hiểu nó. Sẽ không hiệu quả khi xem lại chỉ một phần mã đã viết hoặc mã sẽ được sửa đổi đáng kể bởi lập trình viên.
- **Chức năng tối thiểu:** Mã phải biên dịch và liên kết và được kiểm tra ở một mức độ nhất định nào đó để đảm bảo rằng nó thực hiện các chức năng cơ bản .

- **Khả năng đọc :** Vì việc xem xét mã liên quan đến việc đọc mã thực tế của các lập trình viên khác, điều cần thiết là mã phải có khả năng đọc cao. Một số đặc điểm mã tăng cường khả năng đọc là định dạng thích hợp, sử dụng tên định danh có ý nghĩa, sử dụng dễ dàng các cấu trúc ngôn ngữ lập trình và mức độ trừu tượng thích hợp bằng cách sử dụng các lệnh gọi hàm. Trong trường hợp không có khả năng đọc, những người đánh giá có thể chán nản trong việc thực hiện nhiệm vụ một cách hiệu quả.
- **Độ phức tạp :** Không cần phải lên lịch họp nhóm để xem xét mã đơn giản mà lập trình viên có thể dễ dàng xem xét. Mã được xem xét phải có đủ độ phức tạp để đảm bảo nhóm được xem xét. Ở đây, độ phức tạp là một thuật ngữ tổng hợp đề cập đến số câu lệnh điều kiện trong mã, số phần tử dữ liệu đầu vào của đơn vị, số phần tử dữ liệu đầu ra do đơn vị tạo ra, quá trình xử lý mã theo thời gian thực và số của các đơn vị khác mà mã giao tiếp với nhau.
- **Yêu cầu và Tài liệu thiết kế:** Cần có phiên bản mới nhất được phê duyệt của đặc điểm kỹ thuật thiết kế hoặc các mô tả thích hợp khác về các yêu cầu của chương trình. Các tài liệu này giúp người đánh giá xác minh xem mã đang xem xét có triển khai các chức năng mong đợi hay không , có triển khai thiết kế một cách thích hợp hay không.

### 3.3.2 Chuẩn bị ( Preparation )

Trước cuộc họp, mỗi người đánh giá xem xét cẩn thận gói công việc. Người đánh giá phải đọc mã và hiểu tổ chức và hoạt động của nó trước cuộc họp đánh giá. Mỗi người đánh giá phát triển những điều sau:

- **Danh sách các câu hỏi:** Một người đánh giá chuẩn bị một danh sách các câu hỏi sẽ được hỏi, nếu cần, của tác giả để làm rõ các vấn đề nảy sinh từ việc đọc của họ .
- **Yêu cầu thay đổi ( CR ) tiềm năng:** Người đánh giá có thể đưa ra yêu cầu chính thức để thực hiện thay đổi. Đây được gọi là các yêu cầu thay đổi chứ không phải là báo cáo lỗi. Ở giai đoạn này, vì lập trình viên vẫn chưa công khai mã, nên sẽ thích hợp hơn để đưa ra đề xuất với tác giả để thực hiện các thay đổi, hơn là báo cáo một khiếm khuyết. Mặc dù CR tập trung vào các lỗi trong mã, các báo cáo này không được đưa vào thống kê lỗi liên quan đến sản phẩm.
- **Cơ hội cải tiến được đề xuất:** Người đánh giá có thể đề xuất cách khắc phục các vấn đề, nếu có, trong mã đang xem xét. Vì những người đánh giá là chuyên gia trong lĩnh vực chủ đề của mã, nên không có gì lạ khi họ đưa ra các đề xuất cải tiến.



### 3.3.3 Kiểm tra (Examination)

Quá trình kiểm tra bao gồm các hoạt động sau:

- Tác giả trình bày về logic thủ tục được sử dụng trong mã, các đường dẫn biểu thị các phép tính chính và sự phụ thuộc của đơn vị đang được xem xét vào các đơn vị khác.
- Người thuyết trình đọc từng dòng mã. Người đánh giá có thể đưa ra câu hỏi nếu mã được thấy là có khiếm khuyết. Tuy nhiên, các vấn đề không được giải quyết trong cuộc họp. Người đánh giá có thể đưa ra các đề xuất chung về cách sửa chữa các khiếm khuyết, nhưng việc thực hiện các biện pháp sửa chữa sau khi cuộc họp kết thúc là tùy thuộc vào tác giả .
- Người ghi chép tài liệu các yêu cầu thay đổi và các đề xuất để khắc phục sự cố, nếu có. Một CR bao gồm các chi tiết sau:
  - Đưa ra mô tả ngắn gọn về vấn đề hoặc đề mục hành động.
  - Chỉ định mức độ ưu tiên (chính hoặc phụ) cho CR.
  - Chỉ định một người theo dõi vấn đề. Vì CR ghi lại một vấn đề tiềm ẩn, cần có sự tương tác giữa tác giả của mã và một trong những người đánh giá, có thể là người đánh giá đã thực hiện CR.
  - Đặt thời hạn để giải quyết CR.
- Người điều hành đảm bảo rằng cuộc họp vẫn tập trung vào quá trình xem xét. Người điều hành đảm bảo rằng cuộc họp đạt được tiến độ ở một tốc độ nhất định để đạt được mục tiêu của cuộc họp.
- Vào cuối cuộc họp, một quyết định được đưa ra về việc có nên gọi một cuộc họp khác để xem xét thêm quy tắc hay không. Nếu quá trình xem xét dẫn đến việc làm lại toàn bộ quy tắc hoặc các vấn đề quan trọng được xác định trong quá trình này, thì một cuộc họp khác thường được triệu tập. Nếu không, cuộc họp thứ hai sẽ không được lên lịch và tác giả được giao trách nhiệm sửa các CR .

### 3.3.4 Thực hiện lại ( Rework )

Vào cuối cuộc họp, người ghi chép bản tóm tắt cuộc họp bao gồm các thông tin sau:

- Danh sách tất cả các CR, ngày sửa những CR và tên của những người chịu trách nhiệm xác thực CR
- Danh sách các cơ hội cải tiến
- Biên bản cuộc họp (tùy chọn)

Một bản sao của báo cáo được phân phát cho tất cả các thành viên của nhóm đánh giá.

Sau cuộc họp, Tác giả ghi lại những cải tiến được thực hiện đối với mã trong các CR và cố gắng giải quyết các vấn đề trong khung thời gian đã thỏa thuận bằng cách sử dụng các quy ước mã hóa phổ biến.

### 3.3.5 Kiểm duyệt ( Validation )

CR được xác nhận độc lập bởi người kiểm duyệt hoặc một người khác được chỉ định cho mục đích này.

Quá trình xác thực bao gồm việc kiểm tra mã đã sửa đổi như được ghi trong CR và đảm bảo rằng các cải tiến được đề xuất đã được triển khai chính xác.

Phiên bản sửa đổi và cuối cùng của kết quả của cuộc họp đánh giá được phân phối cho tất cả các thành viên trong nhóm.

### 3.3.6 Thoát ( Exit )

Tóm tắt quá trình xem xét, được cho là hoàn tất nếu tất cả các hành động sau đã được thực hiện:

- Mọi dòng mã trong thiết bị đã được kiểm tra.
- Nếu tìm thấy quá nhiều khiếm khuyết trong một mô-đun, mô-đun đó sẽ được xem xét lại một lần nữa sau khi tác giả áp dụng các hiệu chỉnh. Theo quy tắc chung, nếu hơn 5
- Tác giả và những người đánh giá đạt được sự nhất trí rằng khi áp dụng các hiệu chỉnh, mã sẽ có khả năng không có sai sót.
- Tất cả các CR đều được ghi lại và xác nhận bởi người kiểm duyệt hoặc người khác. Các hành động tiếp theo của tác giả được ghi lại.
- Một báo cáo tóm tắt của cuộc họp bao gồm các CR được phân phát cho tất cả các thành viên của nhóm đánh giá.

Hiệu quả của kiểm thử tĩnh bị hạn chế bởi khả năng của người đánh giá để tìm ra các khiếm khuyết trong mã bằng các phương tiện trực quan.

Tuy nhiên, nếu sự xuất hiện của các lỗi phụ thuộc vào một số giá trị thực tế của các biến, thì việc xác định các lỗi đó bằng các phương tiện trực quan là một nhiệm vụ khó khăn.

Do đó, một đơn vị phải được thực thi để quan sát các hành vi của nó để đáp ứng với nhiều loại đầu vào.

Cuối cùng, bất cứ điều gì có thể là hiệu quả của các bài kiểm tra tĩnh, người ta không thể cảm thấy tự tin nếu không thực sự chạy mã.

### 3.4 Số liệu đánh giá mã nguồn

Điều quan trọng là phải thu thập dữ liệu đo lường liên quan đến quá trình xem xét, để quá trình xem xét có thể được đánh giá, hiển thị cho lãnh đạo cấp trên như một chiến lược thử nghiệm và được cải tiến để có hiệu quả hơn.

Hơn nữa, việc thu thập các số liệu trong quá trình xem xét mã tạo điều kiện thuận lợi cho việc ước tính thời gian xem xét và nguồn lực cho các dự án trong tương lai.

Vì vậy, xem xét mã là một chiến lược kiểm tra khả thi có thể được sử dụng hiệu quả để cải thiện chất lượng của sản phẩm ở giai đoạn đầu.

Các chỉ số sau có thể được thu thập từ việc xem xét mã:

- Số lượng dòng code được khảo sát trong 1 giờ
- Số lượng CR được tạo ra trên 1000 dòng code
- Số lượng CR được tạo ra trên 1 giờ
- Tổng số CR của mỗi project
- Tổng số giờ khảo sát mã nguồn của mỗi project

### 3.5 Danh sách kiểm tra đánh giá mã

1. Mã có thực hiện những gì đã được chỉ định trong đặc điểm kỹ thuật thiết kế không?

2. Quy trình được sử dụng trong mô-đun có giải quyết được vấn đề một cách chính xác không?

3. Mô-đun phần mềm có trùng lặp với một mô-đun hiện có khác có thể được sử dụng lại không?

4. Nếu các mô-đun thư viện đang được sử dụng, có đúng thư viện và phiên bản phù hợp của thư viện đang được sử dụng không?

5. Mỗi mô-đun có một điểm vào duy nhất và một điểm thoát duy nhất không? Khó kiểm tra nhiều chương trình điểm đầu ra và đầu vào hơn.

6. Độ phức tạp theo chu kỳ của mô-đun có lớn hơn 10 không? Nếu có, thì rất khó để kiểm tra đầy đủ mô-đun.

7. Mỗi chức năng nguyên tử có thể được xem lại và hiểu trong 10-15 phút không? Nếu không, nó được coi là quá phức tạp.

8. Các quy ước đặt tên có được tuân theo cho tất cả các định danh, chẳng hạn như con trỏ, chỉ số, biến, mảng và hằng số không? Điều quan trọng là phải tuân thủ các tiêu chuẩn mã hóa để dễ dàng giới thiệu một người đóng góp mới (lập trình viên) cho sự phát triển của hệ thống.

9. Mã đã được nhận xét đầy đủ chưa?
10. Tất cả các biến và hằng đã được khởi tạo chính xác chưa? Đã kiểm tra đúng loại và phạm vi chưa?
11. Các biến toàn cục hoặc biến chia sẻ, nếu có, có được kiểm soát cẩn thận không?
12. Có các giá trị dữ liệu được mã hóa cứng trong chương trình không? Thay vào đó, chúng nên được khai báo dưới dạng các biến.
13. Các con trỏ có được sử dụng đúng cách không?
14. Các khối bộ nhớ có được động có được phân bổ sau khi sử dụng không?
15. Mô-đun có kết thúc bất thường không? Mô-đun cuối cùng sẽ kết thúc?
16. Có khả năng xảy ra vòng lặp vô hạn, vòng lặp không bao giờ thực thi hoặc vòng lặp có lỗi ra sớm không?
17. Tất cả các tệp đã được mở để sử dụng và đóng lại khi kết thúc?
18. Có các phép tính sử dụng các biến có kiểu dữ liệu không nhất quán không? Có khả năng xảy ra tràn hoặc tràn không?
19. Mã lỗi và thông báo điều kiện có được tạo ra bằng cách truy cập một bảng thông báo chung không? Mỗi mã lỗi phải có một ý nghĩa và tất cả các ý nghĩa phải có sẵn ở một nơi trong bảng thay vì nằm rải rác trên toàn bộ mã chương trình.
20. Mã có di động không? Mã nguồn có khả năng thực thi trên nhiều kiến trúc bộ xử lý và trên các hệ điều hành khác nhau trong suốt thời gian tồn tại của nó. Nó phải được thực hiện theo cách không loại trừ nhiều loại môi trường thực thi này.
21. Mã có hiệu quả không? Nói chung, không nên hy sinh tính rõ ràng, dễ đọc hoặc tính đúng đắn để đạt được hiệu quả. Việc xem xét mã nhằm phát hiện các lựa chọn triển khai có ảnh hưởng xấu đến hiệu suất của hệ thống.

## 4 Tài liệu

- [1] Bài giảng lớp Kiểm thử và Đảm bảo chất lượng phần mềm - Cô Nguyễn Thị Thu Trang <https://courses.uet.vnu.edu.vn/course/view.php?id=8435>.
- [2] Software Testing and Quality Assurance : Theory and Practice [https://courses.uet.vnu.edu.vn/pluginfile.php/298194/mod\\_resource/content/1/Software\\_Testing\\_and\\_Quality\\_Assurance\\_T.pdf](https://courses.uet.vnu.edu.vn/pluginfile.php/298194/mod_resource/content/1/Software_Testing_and_Quality_Assurance_T.pdf).