**LAB 5**

* **PHẦN 1**

**Giả sử một ô textbox cho phép nhập số điểm với các ràng buộc như sau: + Số điểm từ 5-14: xếp loại kém**

**+ Số điểm từ 15-40: xếp loại Trung bình**

**+ Số điểm từ 41-60: xếp loại khá**

**+ Số điểm từ 61-100: xếp loại Giỏi**

**Câu 1: Sử dụng kỹ thuật phân vùng tương đương để liệt kê các trường hợp kiểm thử**

Trả lời:

* Nhập số điểm bằng 5
* Nhập số điểm bằng 14
* Nhập số điểm bằng 15
* Nhập số điểm bằng 40
* Nhập số điểm bằng 41
* Nhập số điểm bằng 60
* Nhập số điểm bằng 61
* Nhập số điểm bằng 100

**Câu 2: Sử dụng kỹ thuật phân tích giá trị biên để liệt kê các trường hợp kiểm thử**

Trả lời:

* Nhập số điểm bằng 4 => Fail
* Nhập số điểm bằng 5 => Pass
* Nhập số điểm bằng 14 => Pass
* Nhập số điểm bằng 15 => Pass
* Nhập số điểm bằng 40 => Pass
* Nhập số điểm bằng 41 => Pass
* Nhập số điểm bằng 60 => Pass
* Nhập số điểm bằng 61 => Pass
* Nhập số điểm bằng 100 => Pass
* Nhập số điểm bằng 101 => Fail

**Câu 3:** Sử dụng kỹ thuật bảng quyết định xác định test case cho bài toán khách hàng đến mở thẻ tín dụng với các điều kiện sau:

+ Nếu bạn là một khách hàng mới, đến mở thẻ tín dụng, bạn sẽ được giảm giá 15%.

+ Nếu bạn là khách hàng cũ, và có thẻ Vip, bạn sẽ được giảm giá 10%.

+ Nếu bạn có Coupon, bạn sẽ được giảm giá 20% (nhưng nó không được sử dụng giảm giá cùng với khách hàng mới.

+ Việc giảm giá có thể được cộng nếu như phù hợp.

* Nhập trạng thái khách hàng mới và có coupon => 20%
* Nhập trạng thái khách hàng mới và không có coupon => 15%
* Nhập trạng thái khách hàng cũ và có thẻ Vip và coupon => 30%
* Nhập trạng thái khách hàng cũ và có thẻ Vip nhưng không có coupon => 10%
* Nhập trạng thái khách hàng cũ và không có thẻ Vip và coupon => 0%
* Nhập trạng thái khách hàng cũ và không có thẻ Vip nhưng có coupon => 20%
* **PHẦN 2**

**Câu 4**: Với hàm code bên dưới hãy thực hiện xác định các nút, vẽ đường cơ bản và liệt kê các trường hợp kiểm thử bằng đường đi:

Trả lời:

* 1. Xác định các nút:
* Nút bắt đầu: Đầu tiên, hàm sẽ kiểm tra số nguyên tố cho đầu vào "n".
* Nút điều kiện: Đoạn code "for (int i = 2; i < n; i++)" sẽ dùng để kiểm tra từng số trong khoảng từ 2 đến n-1, nếu có bất kỳ số nào cho ra kết quả chia hết cho "n" thì biến "co" sẽ được gán giá trị false và dừng vòng lặp.
* Nút kết quả: Đoạn code "if (n != 1 && co == true)" sẽ kiểm tra xem biến "co" còn là true hoặc "n" khác 1, nếu điều kiện này đúng, hàm sẽ in ra "là số nguyên tố". Trong trường hợp ngược lại, hàm sẽ in ra "không phải là nguyên tố".
  1. Đường cơ bản:
* Gọi hàm "soNT" với số nguyên "n" như đầu vào.
* Kiểm tra nếu "n" bằng 1, chuyển đến nút đầu ra với "không phải là số nguyên tố".
* Tìm kiếm từ 2 đến "n" - 1 với vòng lặp "for".
* Kiểm tra nếu "n" chia hết cho "i", chuyển đến nút đầu ra với "không phải là số nguyên tố".
* Nếu "n" không chia hết cho bất kỳ "i" nào trong vòng lặp "for", chuyển đến nút đầu ra với "là số nguyên tố".
  1. Các trường hợp kiểm thử:
* Trường hợp nhập "1" cho "n".
* Trường hợp nhập số nguyên tố cho "n".
* Trường hợp nhập số chẵn cho "n".
* Trường hợp nhập số âm cho "n".
* Trường hợp nhập số nguyên tố lớn cho "n".

**Câu 5:** Xác định số test case tối thiểu bao phủ 100% Câu lệnh - Statement coverage  
 if ((temperature < 0) or (temperature > 100))

{ alert ("DANGER");  
 if ((speed > 100) and (load <= 50))

{ speed = 50; } else { check = false; }

Trả lời:

* temperature < 0 và speed > 100 và load <= 50: alert("DANGER"); speed = 50;
* temperature >= 0 và speed <= 100 và load > 50: check = false;
* temperature < 0 và speed <= 100 và load > 50: alert("DANGER"); check = false;
* temperature >= 0 và speed > 100 và load <= 50: alert("DANGER"); speed = 50;

***END***