

1

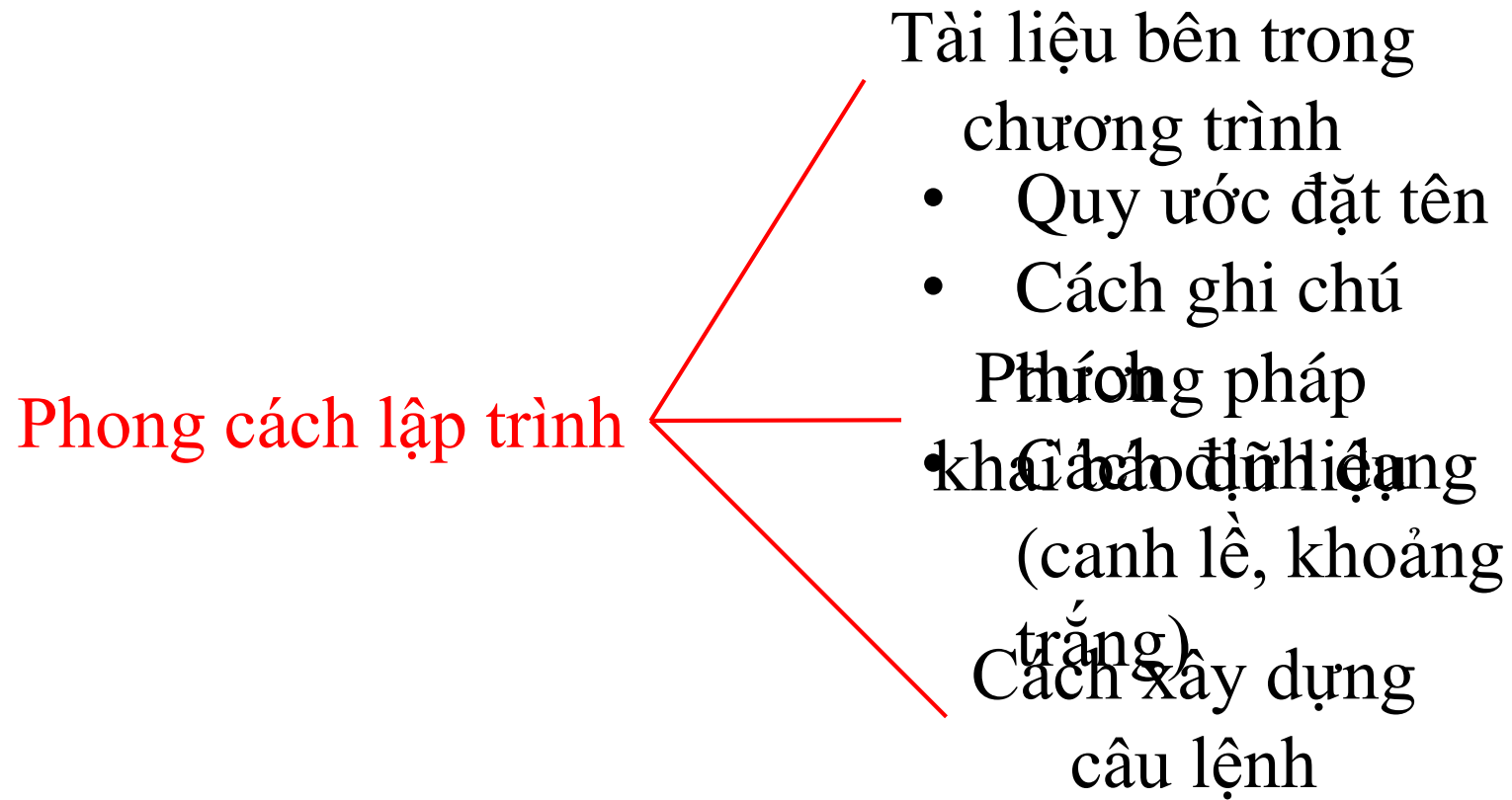
---

# Phong cách lập trình



# Phong cách lập trình

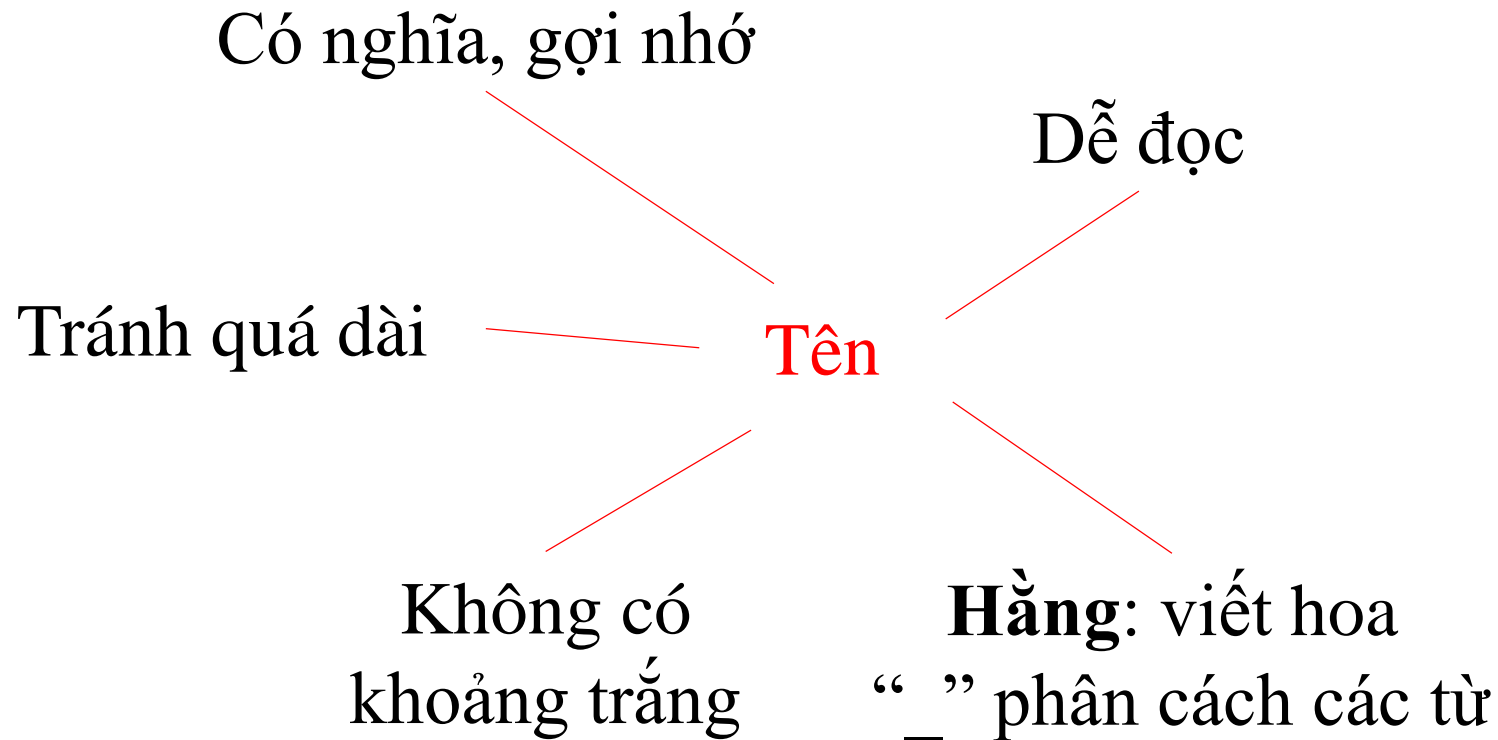
---



<https://www.monash.edu/it/current-students/resources-and-support/style-guide/programming-styles>

# Quy ước đặt tên

---



# Quy ước đặt tên (C#)

---

Loại	Kiểu đặt tên	Ví dụ	Ghi chú
Tên biến	Camel	backColor	Danh từ
Hằng số	Uppercase	NUMBER_OF_STUDENT	Có gạch chân giữa các từ
Tên class, enum	Pascal	SmartSnake	Danh từ
Tham số	Camel	displayTime	Danh từ
Thuộc tính	Pascal	BackColor	Danh từ
Phương thức	Pascal	GetPath()	Động từ
Sự kiện	Pascal	TextChangedEventHandler	Có hậu tố EventHandler
Giao diện (interface)	Pascal	IButtonControl	Có tiền tố I

# Cách ghi chú thích

---

1. Đúng chính tả
2. Đúng kiểu chú thích của ngôn ngữ sử dụng

Thành phần	Miêu tả	Vị trí
File	Nội dung	Đầu
Class	Mục đích và cách sử dụng	Đầu
Function	Mục đích	Đầu
Biến (nếu có)	Ý nghĩa	Tại vị trí khai báo

# Cách định dạng

- Phân tách các **nhóm code logic**:  
**dòng trắng**
- Phân tách các **hàm**: **tối thiểu 2**  
**dòng trắng**
- Thống nhất cách sử dụng tab, khoảng trắng
- Mỗi dòng tối đa 80 ký tự

```
1 a = (b + c) * d;
2 // is more readable than:
3 a=(b+c)*d
4 while (true) {}
5 // is more readable than:
6 while(true){}
7 for (i = 0; i < 10; i++) {}
8 // is more readable than
9 for(i=0;i<10;i++){
10 if (conditionA && conditionB) {
11     doSomething;
12 }
13 else {
14     doNothing;
15 }
16 // is more readable than:
17 if (conditionA&&conditionB)
18 {doSomething;}
19 else{doNothing;}
```

2

---

## Kiểm thử phần mềm



# Nội dung

---

- Kiểm thử phần mềm?
- Mức độ kiểm thử
- Phương pháp kiểm thử
- Nguyên lý kiểm thử

# Kiểm thử phần mềm?

---

- Y2K
- WannaCry (2017)
- Boeing 737 MAX 8 (2018-2019)

Là tổ chức vận hành phần mềm một cách có kế hoạch  
và phương pháp để tìm ra lỗi

# Mức độ kiểm thử

---

Các yêu cầu nghiệp vụ

Thiết kế mức cao

Thiết kế mức

Tạo mã

Kiểm thử chấp nhận

Kiểm thử hệ thống

Kiểm thử tích hợp

Kiểm thử đơn vị

Đáp ứng yêu cầu?

- Kiểm thử alpha
- Kiểm thử beta

Tổng thể

Một bộ phận chức năng/hệ con

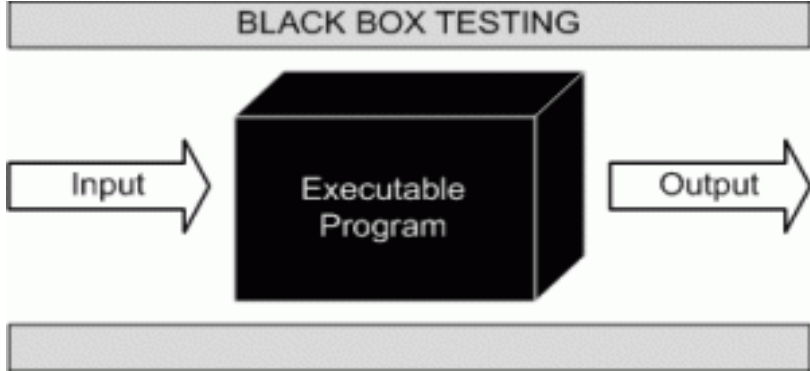
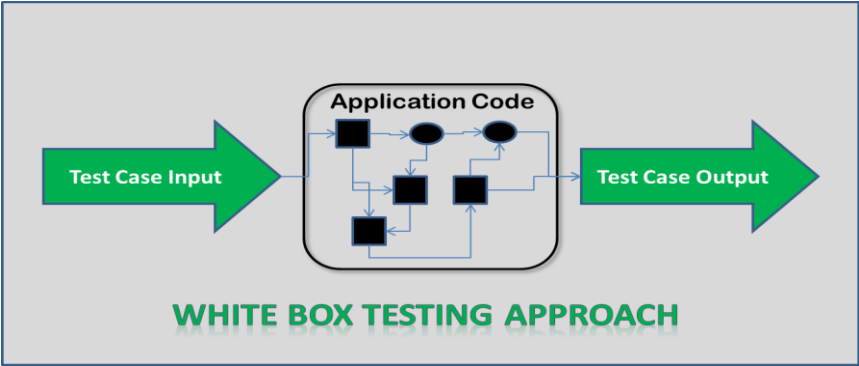
Các module đơn vị chương trình

# Giới hạn của kiểm thử

---

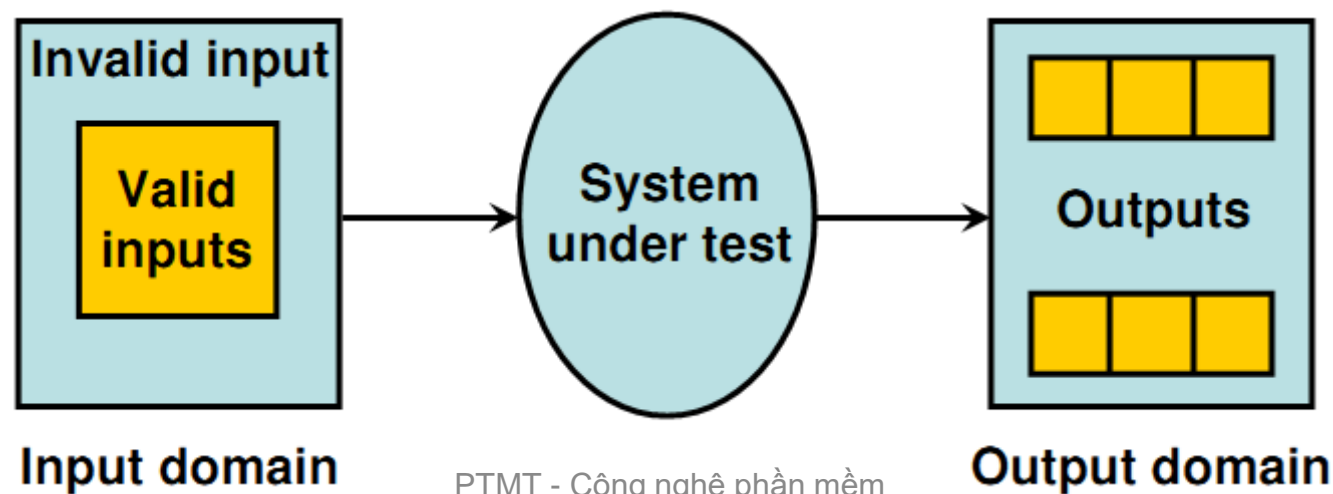
- Miền đầu vào quá lớn
- Có nhiều đường đi trong chương trình
- Giao diện người dùng phức tạp

# Phương pháp kiểm thử

Kiểm thử hộp đen/Kiểm thử chức năng	Kiểm thử hộp trắng/Kiểm thử cấu trúc
	
Không thấy được mã nguồn	Thấy được mã nguồn
Ca kiểm thử độc lập với chương trình cài đặt	Không
Chi phí thấp	Chi phí cao
Áp dụng được cho tất cả các mức độ kiểm thử	Chủ yếu áp dụng cho kiểm thử đơn vị, tích hợp
Hiệu quả tìm lỗi cao: phân tích, thiết kế, cài đặt	Hiệu quả tìm lỗi cài đặt

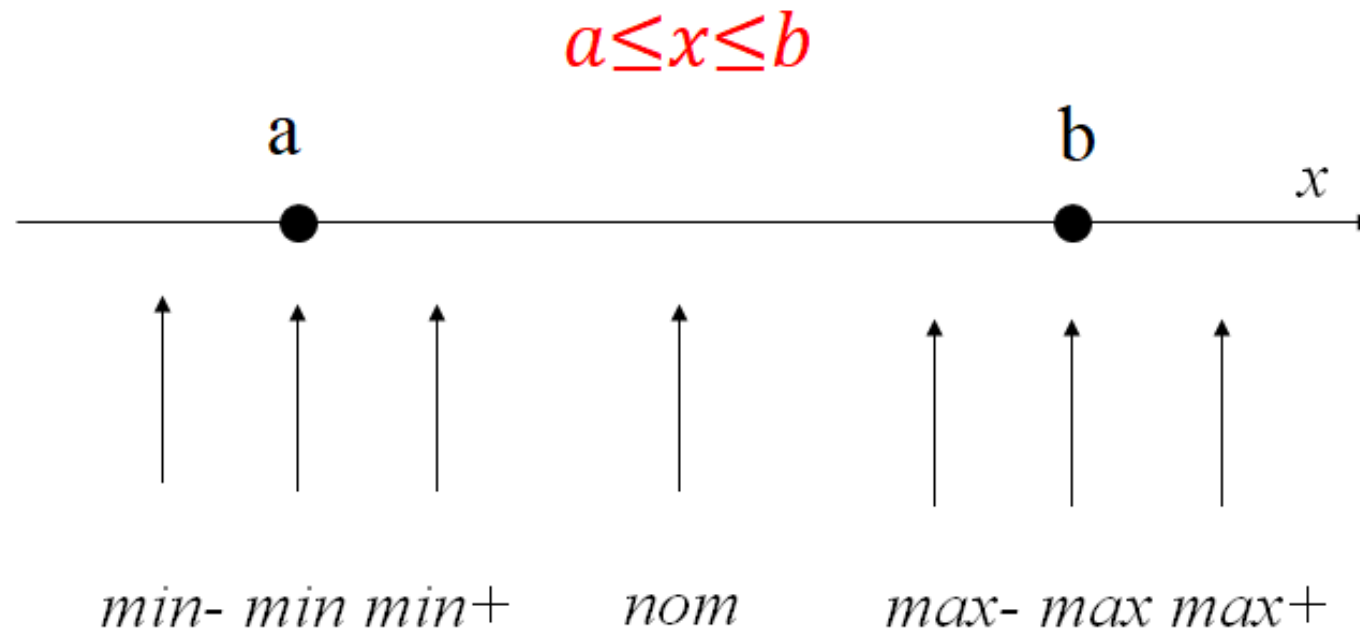
# Phương pháp kiểm thử hộp đen

- Kiểm thử giá trị biên
- Kiểm thử lớp tương đương
  - Chia miền dữ liệu đầu vào và đầu ra thành các lớp tương đương
  - Mỗi lớp hoặc đúng hoặc sai, chỉ cần kiểm tra một số giá trị đặc trưng của nó → rút được ca kiểm thử



# Phương pháp kiểm thử hộp đen

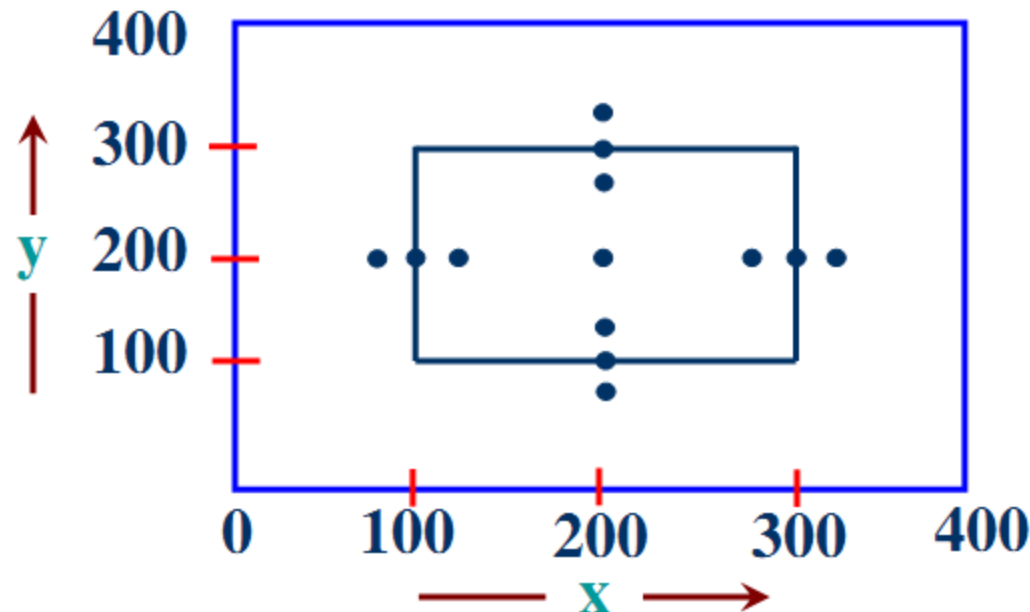
---



# Phương pháp kiểm thử hộp đen

Xem xét chương trình với 2 biến vào x và y, với:  $a \leq x \leq b, c \leq y \leq d$

- Giả sử (x, y) nằm trong đoạn [100, 300]
- Xét lỗi đơn



(200, 99), (200, 100), (200, 101), (200, 200), (200, 299), (200, 300), (200, 301), (99, 200), (100, 200), (101, 200), (299, 200), (300, 200), (301, 200)



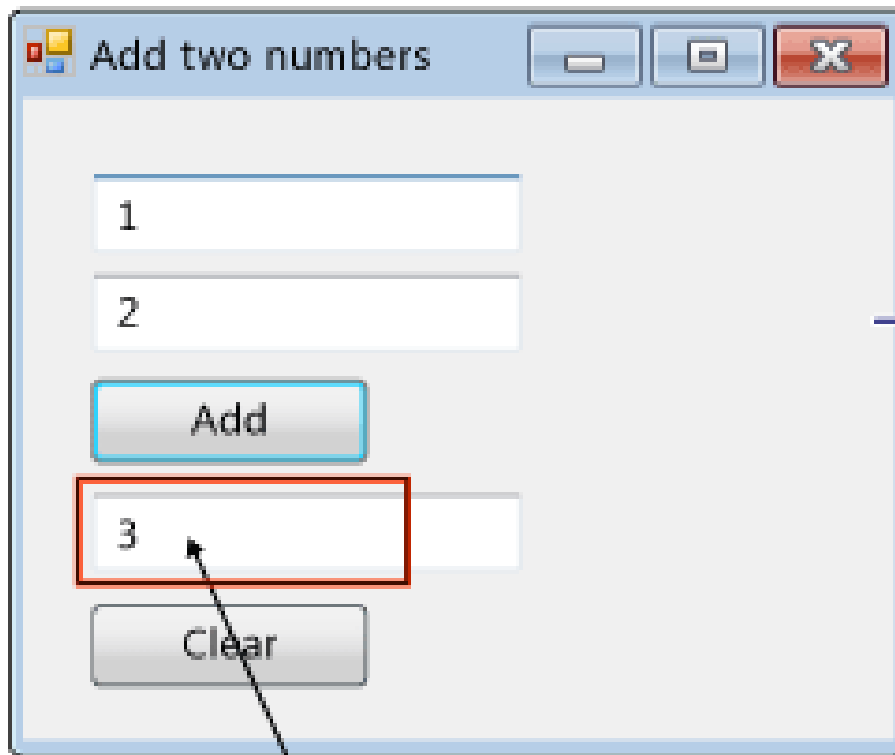
# Phương pháp kiểm thử hộp đen

---

- Cho biết điểm và xếp loại như sau:
  - [5, 15): Kém
  - [15, 40): Trung bình
  - [40, 60): Khá
  - [60, 100): Giỏi
- Hỏi: có mấy lớp tương đương? Giá trị biên?

# Phương pháp kiểm thử hộp đen

- Thực hiện kiểm tra màn hình sau:



màn hình thực hiện  
cộng hai số nguyên dương

kết quả

# Nguyên tắc kiểm thử

---

1. Testing shows presence of defects - Kiểm thử đưa ra lỗi
2. Exhaustive testing is impossible - Kiểm thử đầy đủ là không thể
3. Early testing - Kiểm thử sớm
4. Defect clustering - Sự tập trung của lỗi
5. Pesticide paradox - Nghịch lý thuốc trừ sâu
6. Testing is context dependent - Kiểm thử theo ngữ cảnh độc lập
7. Absence-of-errors fallacy - Sự sai lầm về việc không có lỗi

---

Do the things right >< Do the right things

# Thảo luận

---

- Khi không phát hiện lỗi trong quá trình kiểm thử, theo anh/chị có thể khẳng định chương trình đúng 100% không? Giải thích.

# Hỏi & đáp

---

