



BÀI 9

LẬP TRÌNH VÀ NGÔN NGỮ LẬP TRÌNH

Trường Công nghệ thông tin và Truyền thông
2023

Nội dung

1. Chương trình và Lập trình
2. Các bước lập trình
3. Ngôn ngữ lập trình

1. Chương trình và Lập trình

- Chương trình (Program) là dãy các lệnh mà máy tính thực hiện theo để hoàn thành nhiệm vụ xử lý dữ liệu thành thông tin.
- Lập trình (Programming) hay phát triển phần mềm là thủ tục gồm các bước để tạo ra chương trình.

2. Các bước lập trình

Bước 1: Đặc tả chương trình (Program Specification)

Bước 2: Thiết kế chương trình (Program design)

Bước 3: Viết mã chương trình (Program code)

Bước 4: Kiểm thử chương trình (Program test)

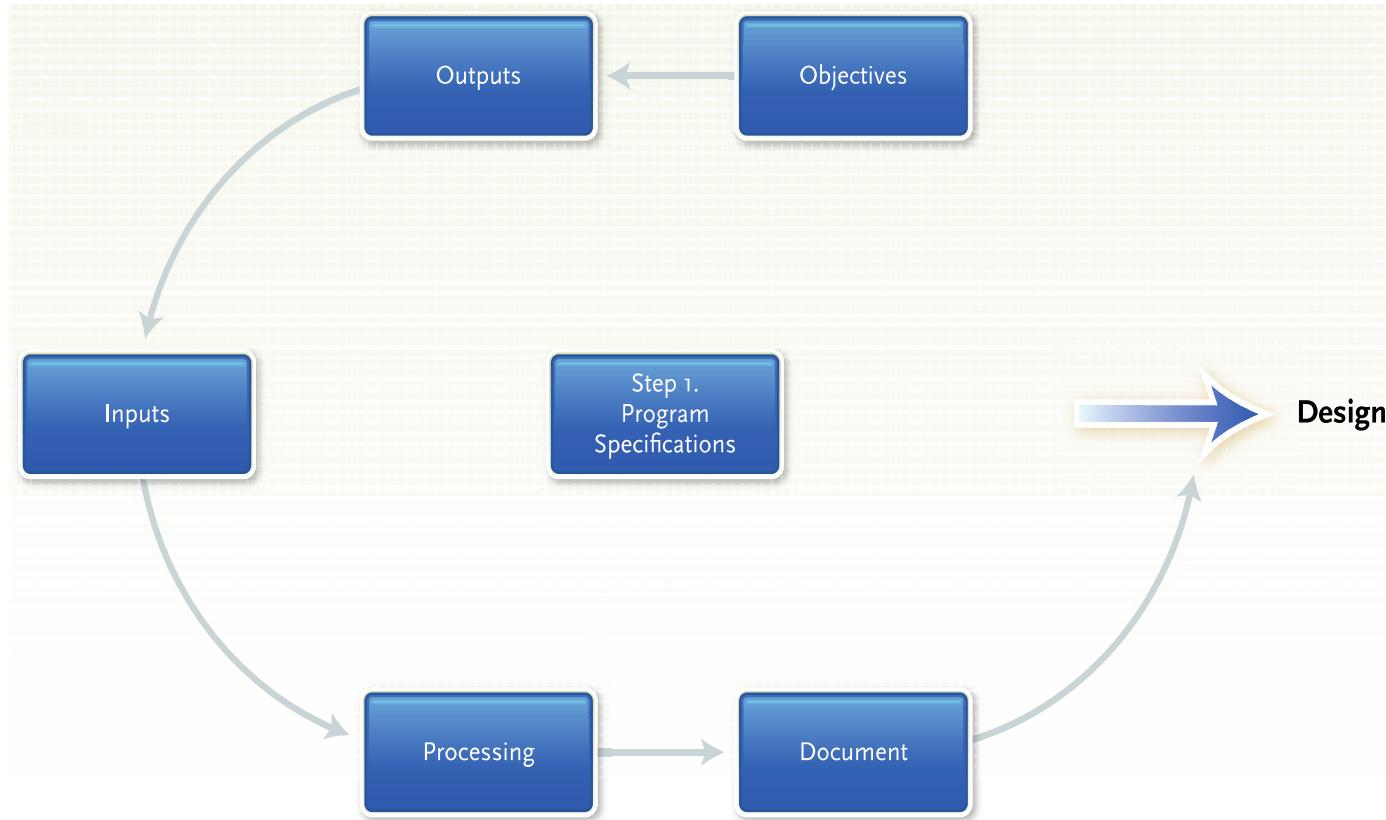
Bước 5: Lập tư liệu chương trình (Program documentation)

Bước 6: Bảo trì chương trình (Program maintenance)

Bước 1. Đặc tả chương trình

- Còn được gọi là phân tích chương trình
- Các việc cần làm:
 - Xác định các mục tiêu của chương trình (các vấn đề cần giải quyết)
 - Xác định các đầu ra muốn có
 - Xác định các dữ liệu đầu vào cần có
 - Xác định các yêu cầu xử lý
 - Lập tư liệu đặc tả chương trình

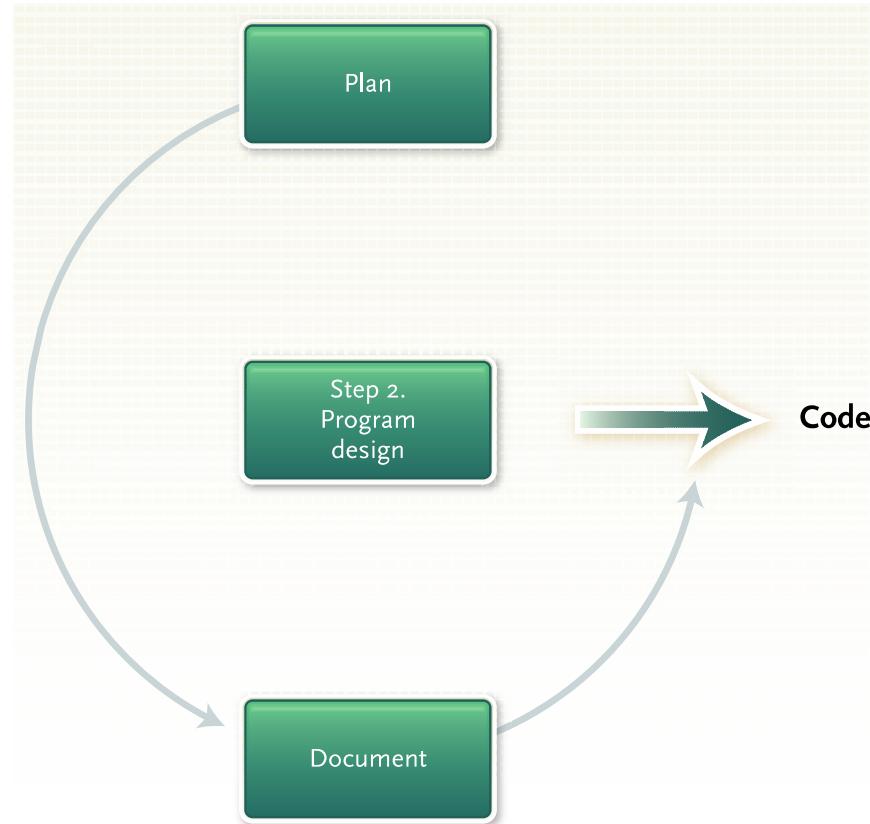
Bước 1. Đặc tả chương trình



Bước 2: Thiết kế chương trình

- Lập kế hoạch giải vấn đề/bài toán sử dụng kỹ thuật lập trình cấu trúc:
 - Thiết kế top-down: Xác định các bước xử lý chính, các modul chương trình sẽ được gọi
 - Mã giả (pseudocode): Mô tả cách giải bài toán theo ngôn ngữ tự nhiên
 - Lưu đồ chương trình (Flowcharts)
 - Các cấu trúc logic
 - Các thuật toán liên quan
- Viết tài liệu thiết kế

Bước 2: Thiết kế chương trình



Thuật giải (Algorithm)

- Thuật giải là dãy các lập luận và thao tác cung cấp lời giải của một vấn đề, một bài toán.
- Các tính chất của thuật giải
 - Tính chính xác: để đảm bảo kết quả tính toán hay các thao tác mà máy tính thực hiện được là chính xác.
 - Tính rõ ràng: Thuật toán phải được thể hiện bằng các câu lệnh minh bạch; các câu lệnh được sắp xếp theo thứ tự nhất định.
 - Tính khách quan: Một thuật toán dù được viết bởi nhiều người trên nhiều máy tính vẫn phải cho kết quả như nhau.
 - Tính phổ dụng: Thuật toán không chỉ áp dụng cho một bài toán nhất định mà có thể áp dụng cho một lớp các bài toán có đầu vào tương tự nhau.
 - Tính kết thúc: Thuật toán phải gồm một số hữu hạn các bước tính toán.

Các ký hiệu dùng cho Lưu đồ (Flowchart)



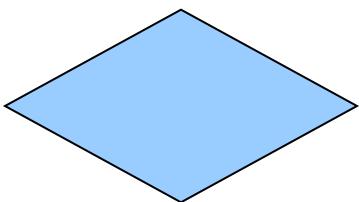
Đầu cuối (Terminal): chỉ ra bắt đầu và kết thúc chương trình



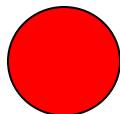
Tiến trình (Process): Tính toán hoặc gán giá trị cho biến



Nhập/Xuất dữ liệu (Input/Output)



Quyết định dựa theo điều kiện (Decision)



Điểm kết nối

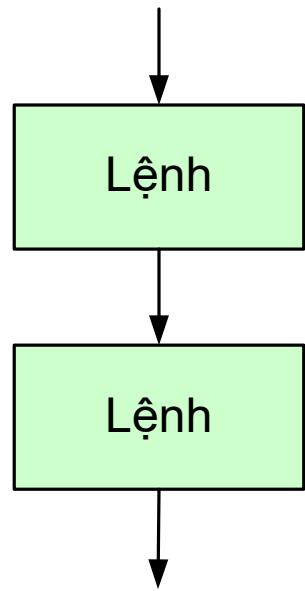


Đường kết nối và chỉ trích tự thao tác

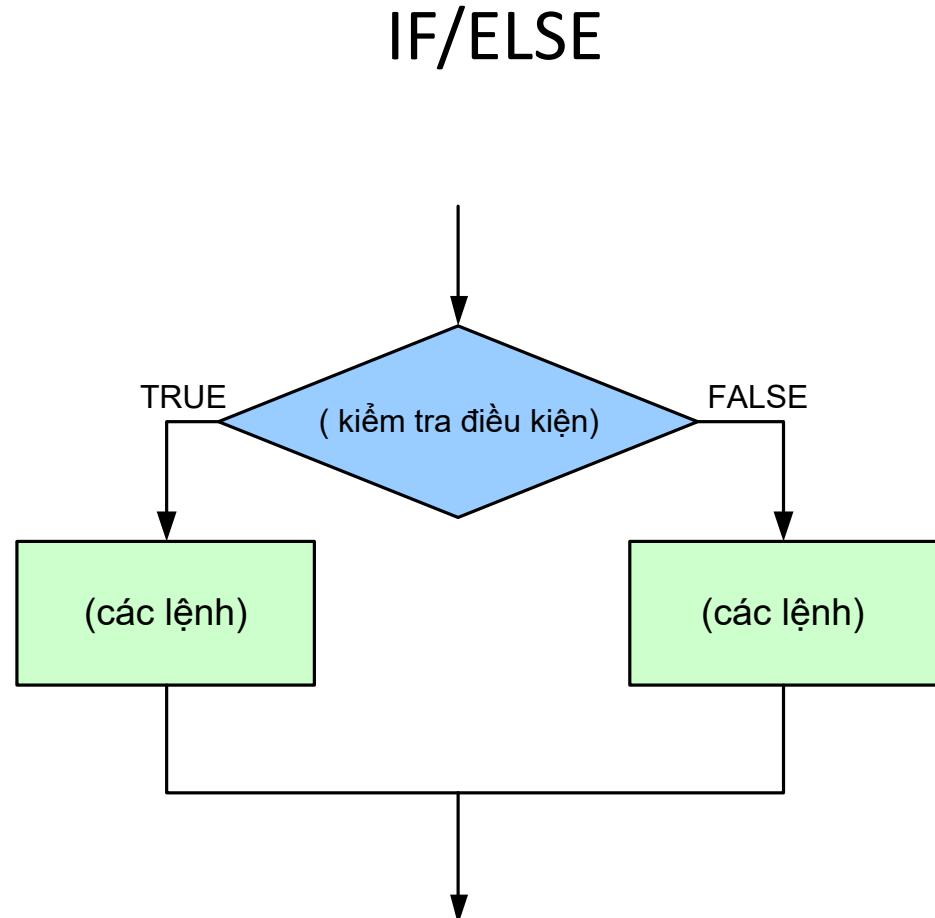
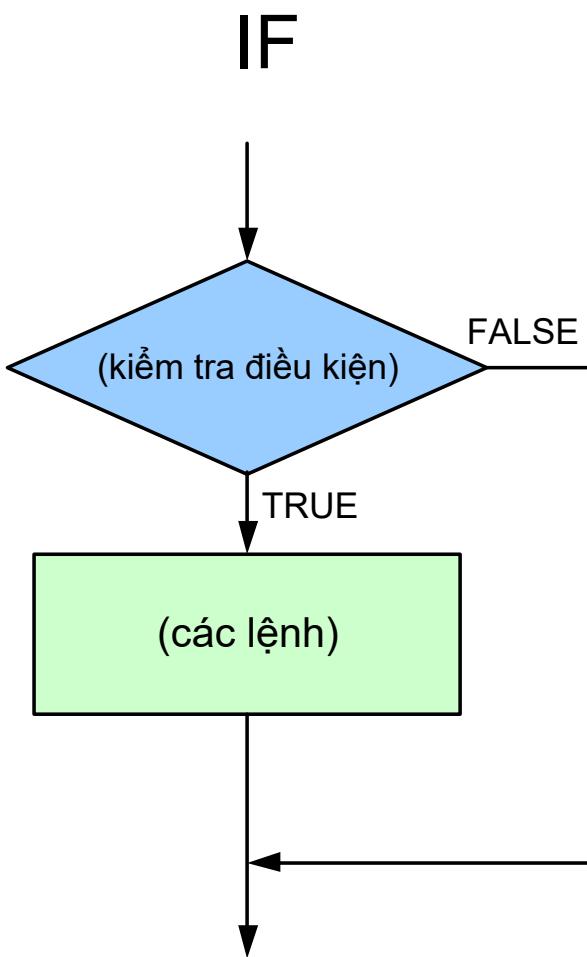
Các cấu trúc logic

- Cấu trúc tuần tự
- Cấu trúc điều khiển lựa chọn:
 - IF
 - IF/ELSE
 - SWITCH ...CASE
- Cấu trúc điều khiển lặp:
 - FOR
 - WHILE
 - DO...WHILE

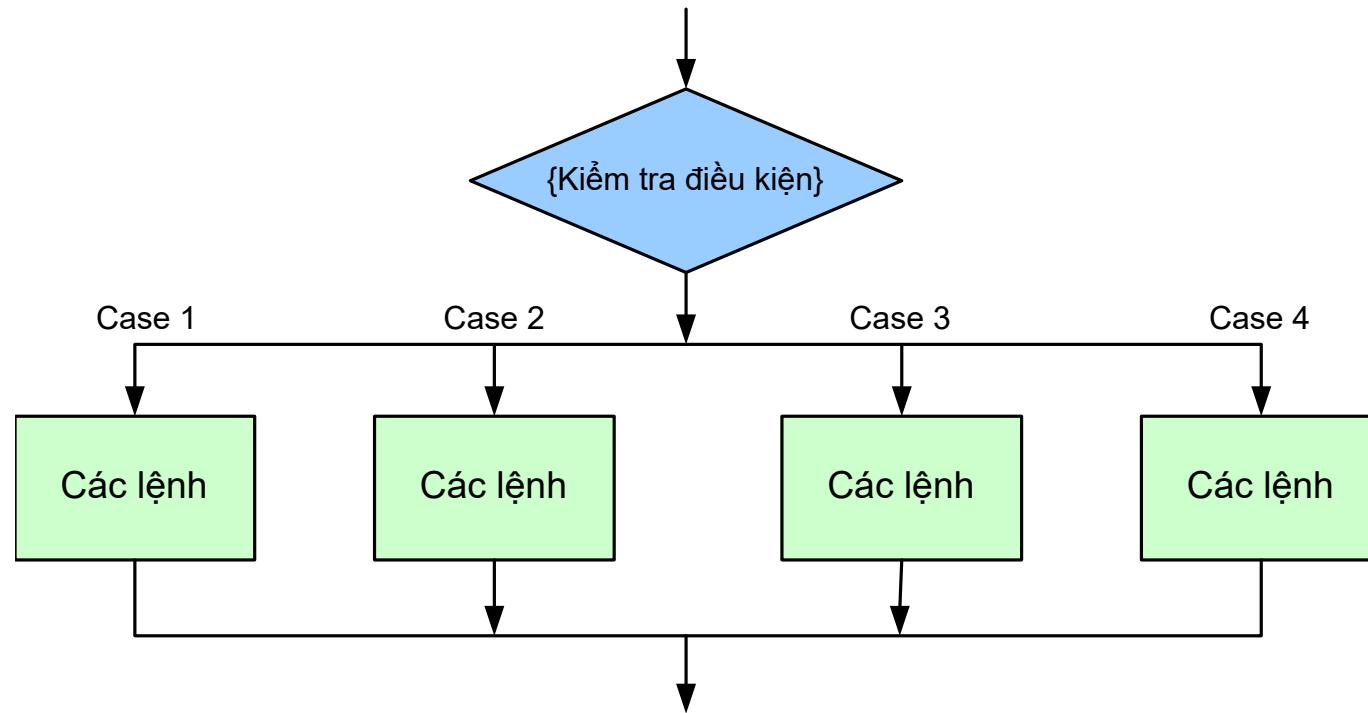
Cấu trúc tuần tự



Các cấu trúc lựa chọn

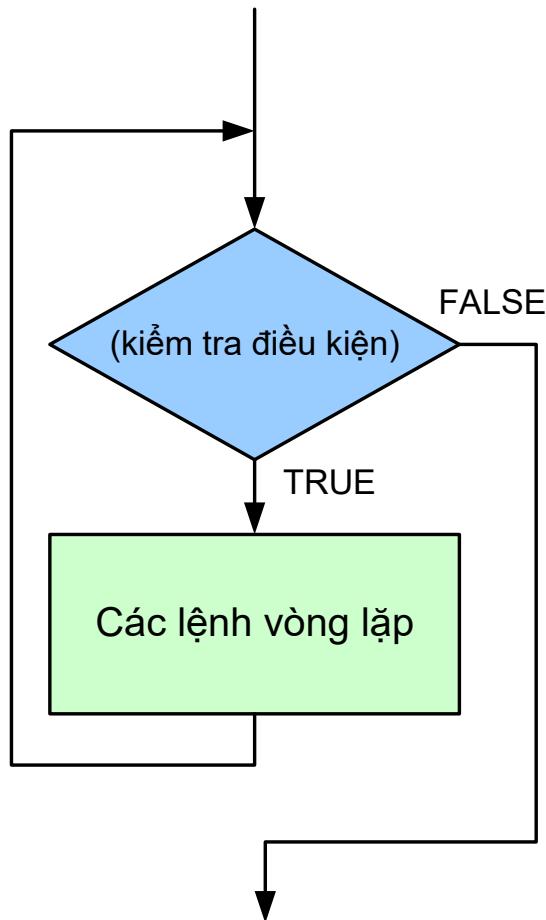


Cấu trúc nhiều lựa chọn SWITCH...CASE

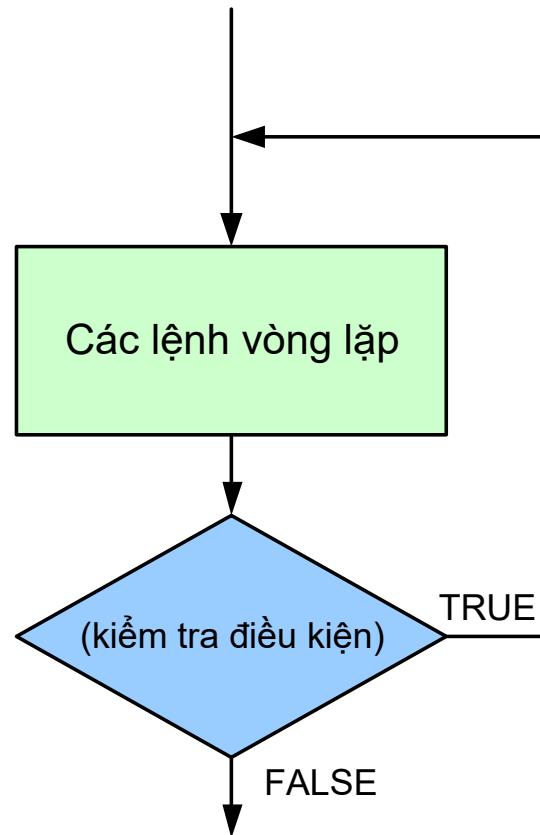


Cấu trúc lặp có số lần lặp không xác định

WHILE



DO...WHILE



Cấu trúc lặp có số lần lặp xác định

- Cấu trúc FOR
- Các lệnh của vòng lặp được thực hiện với số lần lặp biết trước
- Có thể coi cấu trúc FOR tương đương với cấu trúc WHILE hoặc DO-WHILE

Ví dụ mã giả (pseudocode)

START

Khởi tạo Tổng tiền = 0

WHILE (còn hàng trong giỏ)

 Nhập Tên mặt hàng

 Nhập Giá mặt hàng

 Nhập Số lượng của mặt hàng

 Tiền mặt hàng = Giá mặt hàng * Số lượng

 IF (Giá mặt hàng >= 100.000 đồng)

 THEN Thuế = Tiền mặt hàng * 10%

 ELSE Thuế = Tiền mặt hàng * 12%

 ENDIF

 Tiền mặt hàng sau thuế = Tiền mặt hàng + Thuế

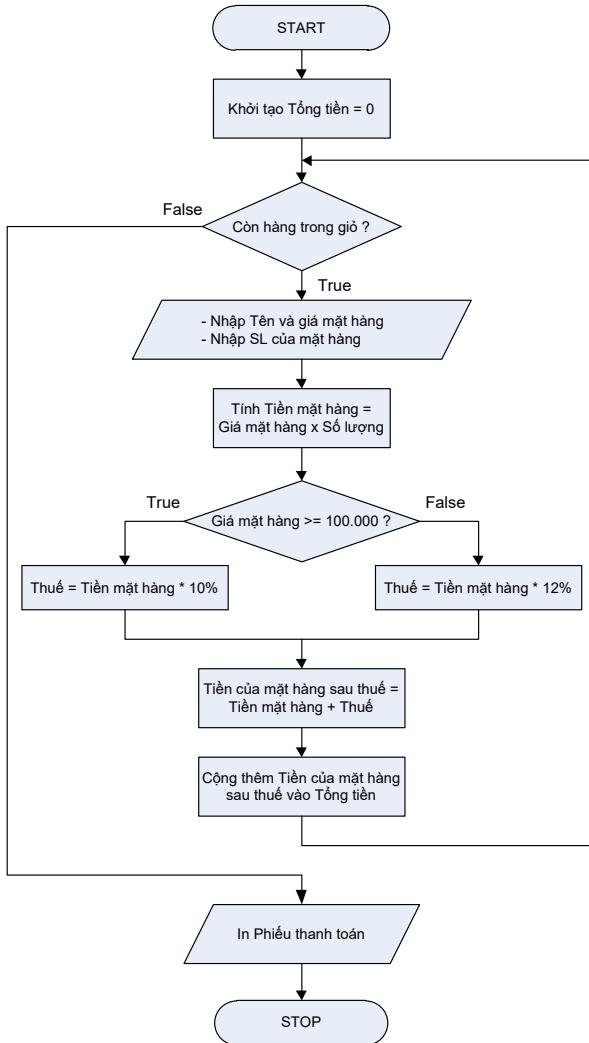
 Cộng Tiền mặt hàng sau thuế vào Tổng tiền

ENDWHILE

In Phiếu thanh toán

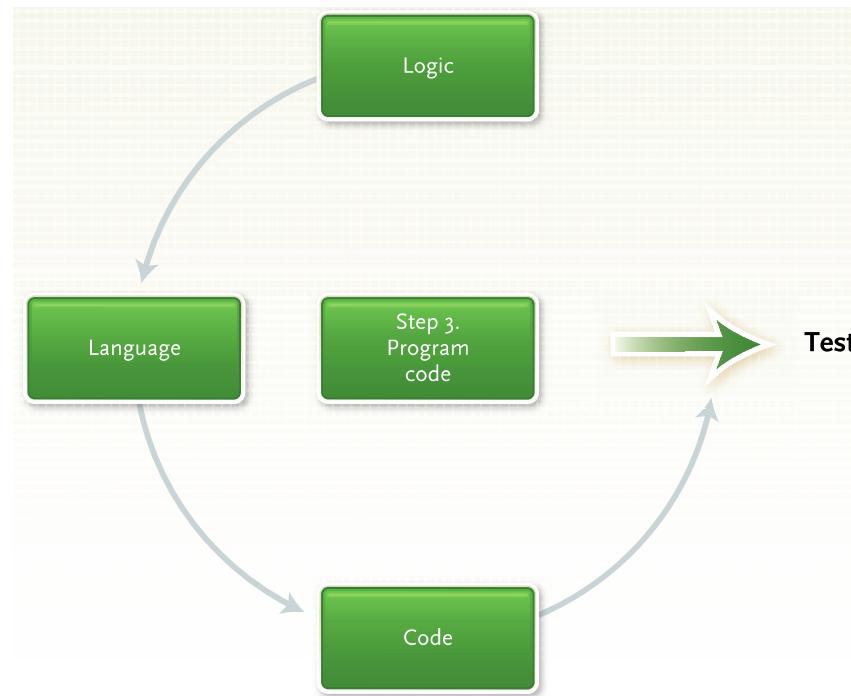
END

Ví dụ lưu đồ



Bước 3: Viết mã chương trình

- Chọn ngôn ngữ lập trình thích ứng
- Viết mã chương trình theo cú pháp



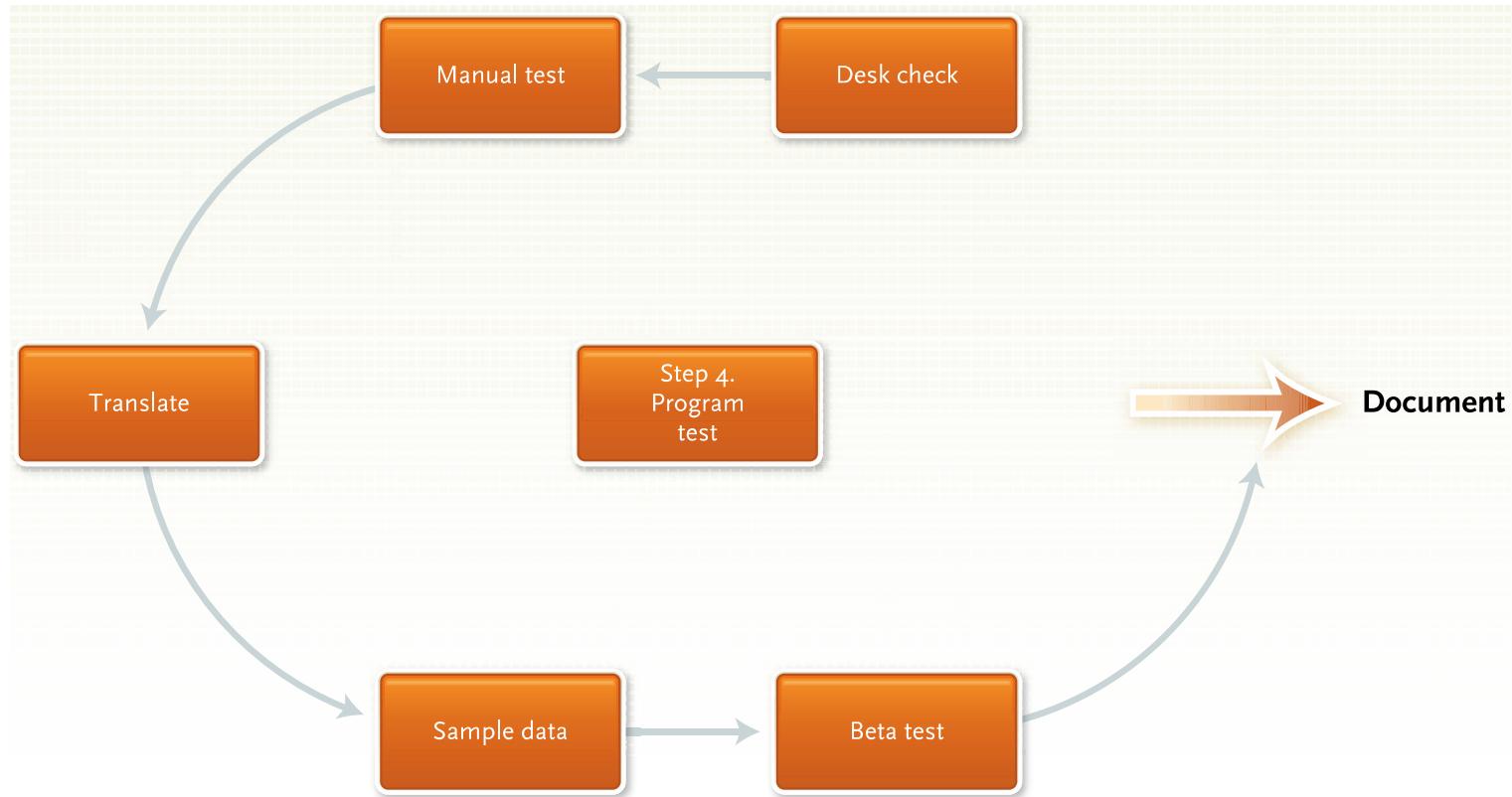
Các đặc trưng của một chương trình tốt

- Làm việc tin cậy
- Tạo đầu ra chính xác
- Bắt được các lỗi đầu vào
- Mã chương trình dễ hiểu và được giải thích rõ ràng
- Sử dụng ngôn ngữ lập trình phù hợp

Bước 4: Kiểm thử chương trình

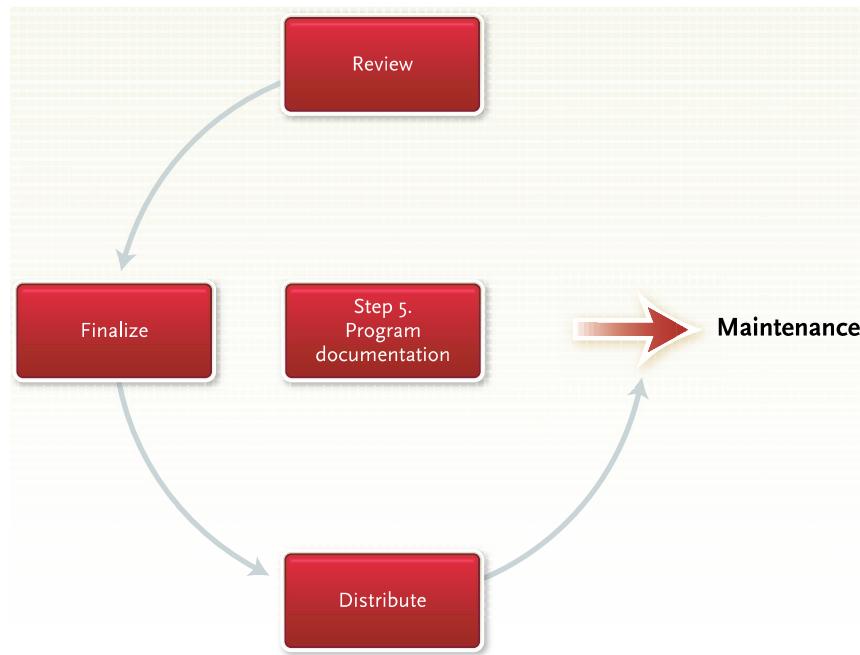
- Chạy thử và phát hiện lỗi:
 - Lỗi cú pháp
 - Lỗi logic
- Chạy chương trình và gỡ rối (alpha testing)
- Chạy chương trình với dữ liệu thực (beta testing)

Bước 4: Kiểm thử chương trình



Bước 5: Lập tư liệu chương trình

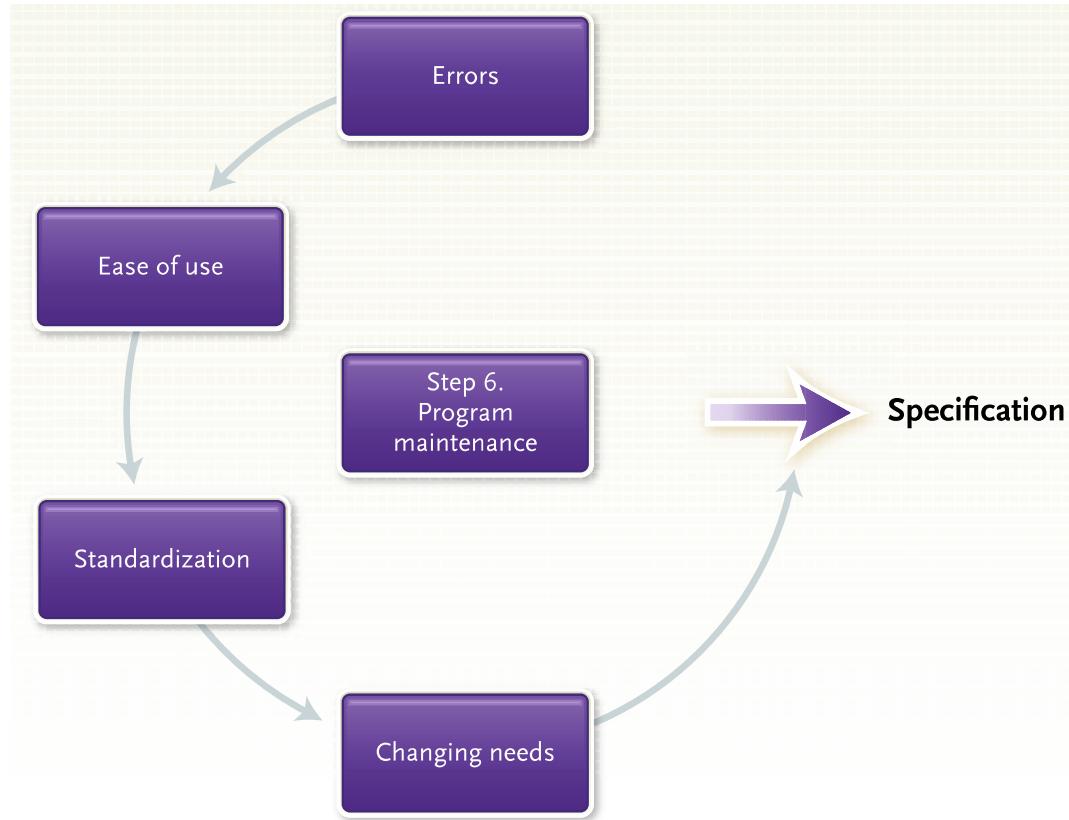
- Viết tư liệu cho người sử dụng
- Viết tư liệu cho người vận hành
- Viết tư liệu cho người lập trình



Bước 6: Bảo trì chương trình

- Đảm bảo để chương trình chạy chính xác, năng suất và hiệu quả.
- Hai tiêu chuẩn bảo trì chương trình:
 - Xác định và hiệu chỉnh các lỗi, làm cho chương trình dễ sử dụng và chuẩn hoá mã chương trình.
 - Có những thay đổi cần thiết theo thời gian.

Bước 6: Bảo trì chương trình



CASE Tools

- CASE Tools (Computer-Aided Software Engineering): được sử dụng để tự động hóa và hỗ trợ cho việc thiết kế, viết mã và kiểm thử chương trình.

Lập trình hướng đối tượng - OOP

- OOP (Object-Oriented Programming)
- Dữ liệu và các lệnh xử lý các dữ liệu đó được nhóm lại thành một đối tượng độc lập để có thể sử dụng lại trong các chương trình khác.
 - Đóng gói
 - Tính thừa kế
 - Tính đa hình

3. Ngôn ngữ lập trình

- Các thể hệ ngôn ngữ lập trình:
 - Ngôn ngữ máy (Machine languages)
 - Hợp ngữ (Assembly languages)
 - Ngôn ngữ thủ tục bậc cao (High-level procedural languages)
 - Ngôn ngữ hướng nhiệm vụ (Task-oriented languages)
 - Ngôn ngữ vấn đề và ràng buộc (Problem and constraint languages)

Ngôn ngữ máy

- Là ngôn ngữ cơ sở của máy tính
- Được biểu diễn bằng các bit 0 và 1

Hợp ngữ

- Ngôn ngữ lập trình bậc thấp
- Chương trình dịch hợp ngữ thành ngôn ngữ máy → assembler

Ngôn ngữ thủ tục bậc cao

- Ngôn ngữ thủ tục
- FORTRAN, COBOL, BASIC, Pascal, C
- Chương trình dịch:
 - Chương trình biên dịch (Compiler): dịch xong toàn bộ chương trình thành ngôn ngữ máy, sau đó máy tính thực hiện chương trình.
 - Chương trình thông dịch (Interpreter): dịch từng lệnh thành ngôn ngữ máy và thực hiện luôn.

Ngôn ngữ hướng nhiệm vụ

- Ngôn ngữ được thiết kế để giải các vấn đề xác định
- Thường được sử dụng trong các hệ quản trị cơ sở dữ liệu
- Các loại:
 - Ngôn ngữ truy vấn (Query Languages)
 - Tạo ứng dụng (Application Generators)
- Ví dụ: SQL

Ngôn ngữ vấn đề và ràng buộc

- Ngôn ngữ kết hợp các khái niệm của trí tuệ nhân tạo cho phép con người cung cấp bối cảnh và các ràng buộc cho hệ thống, sau đó yêu cầu giải quyết
- Cho phép con người trao đổi thông tin trực tiếp với máy tính bằng ngôn ngữ tự nhiên.

Ngôn ngữ máy

- Là ngôn ngữ cơ sở của máy tính
- Được biểu diễn bằng các bit 0 và 1

Hợp ngữ

- Ngôn ngữ lập trình bậc thấp
- Chương trình dịch hợp ngữ thành ngôn ngữ máy → assembler

Ngôn ngữ thủ tục bậc cao

- Ngôn ngữ thủ tục
- FORTRAN, COBOL, BASIC, Pascal, C
- Chương trình dịch:
 - Chương trình biên dịch (Compiler): dịch xong toàn bộ chương trình thành ngôn ngữ máy, sau đó máy tính thực hiện chương trình.
 - Chương trình thông dịch (Interpreter): dịch từng lệnh thành ngôn ngữ máy và thực hiện luôn.

Ngôn ngữ hướng nhiệm vụ

- Ngôn ngữ được thiết kế để giải các vấn đề xác định
- Thường được sử dụng trong các hệ quản trị cơ sở dữ liệu
- Các loại:
 - Ngôn ngữ truy vấn (Query Languages)
 - Bộ tạo ứng dụng (Application Generators)
- Ví dụ: SQL

Ngôn ngữ vấn đề và ràng buộc

- Ngôn ngữ kết hợp các khái niệm của trí tuệ nhân tạo cho phép con người cung cấp bài toán và các ràng buộc cho hệ thống, sau đó yêu cầu giải quyết.
- Cho phép con người trao đổi thông tin trực tiếp với máy tính bằng ngôn ngữ tự nhiên.

Một số ngôn ngữ lập trình cơ bản

- BASIC
- Pascal
- C
- Ngôn ngữ lập trình hướng đối tượng:
 - C++
 - Java

Các ngôn ngữ đánh dấu

- HTML (Hyper Text Markup Language): được sử dụng để tạo các trang Web
- VRML (Virtual Reality Modeling/Markup Language): Tạo Web 3 chiều (3D) và hoạt hình tương tác.
- XML (eXtensible Markup Language): hỗ trợ chia sẻ dữ liệu trên mạng
- XHTML (eXtended HTML): kết hợp HTML và XML để thêm cấu trúc và mềm dẻo cho HTML

Một số ngôn ngữ Script

- Java Script: cho các trang Web động
- Flash: tạo hoạt hình, nhúng audio và video vào trang Web
- ActiveX: Tạo các trang Web tương tác
- PHP: tạo nội dung động trên trang web tương tác với cơ sở dữ liệu.
- ...