



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÀI GIẢNG MÔN

TIN HỌC CƠ SỞ 2

CHƯƠNG MỞ ĐẦU-TỔNG QUAN



Giảng viên:

Điện thoại/E-mail:

Bộ môn:

TS. Nguyễn Quý Sỹ

091 3394091/synq@ptit.edu.vn

Trung tâm Thí nghiệm Thực hành

NỘI DUNG

1. Tổng quan về môn học
2. Tổng quan về ngôn ngữ lập trình C/C++
3. Lưu đồ thuật toán
4. Bài tập về thuật toán

1. Tổng quan về môn học Tin học cơ sở 2

Mục tiêu của môn học

❖ Về kiến thức:

- Các khái niệm cơ bản về ngôn ngữ lập trình
- Các lệnh vào ra và lệnh điều khiển
- Hàm và chương trình
- Hàm và con trỏ

❖ Kỹ năng:

- Soạn thảo, dịch, sửa lỗi và chạy chương trình.
- Tổ chức các hàm, chương trình.
- Lập trình cho các bài toán kỹ thuật thông thường
- Nâng cao khả năng làm việc nhóm (cộng tác, thảo luận...) giữa các thành viên

❖ Thái độ, Chuyên cần:

- Sinh viên có ý thức ứng dụng CNTT để giải quyết các bài toán kỹ thuật trong thực tế.
- Sinh viên tham dự đầy đủ các tiết học lý thuyết, thực hành và làm bài tập

1. Tổng quan về môn học Tin học cơ sở 2

❖ Về thời lượng môn học:

- Tổng số tiết: 2 tín chỉ = 30 tiết
- Lý thuyết: 12 kíp x 2 tiết = 24 tiết
- Thực hành: 2 buổi x 4 tiết = 8 tiết

❖ Đánh giá:

- Chuyên cần: 10%
- Kiểm tra 1 tiết: 10%
- Bài tập lớn: 10% (Làm ít hơn 50 bài/230 bài sẽ không được thi).
- Thi học kỳ: 70%

1. Tổng quan về môn học Tin học cơ sở 2

❖ Tài liệu học tập bắt buộc:

- Bài giảng “Tin học cơ sở 2”, Phan Thị Hà, 2013

❖ Tài liệu học tập tham khảo:

- Tin học đại cương
- Lập trình C/C++
- https://vietjack.com/lap_trinh_c/
- Google.com Tìm kiếm “Lập trình C”

1. Tổng quan về môn học Tin học cơ sở 2

❖ Công cụ hỗ trợ lập trình và bài tập:

- **Website: <http://code.ptit.edu.vn>**
 - Account: Mã sinh viên
 - Pass: Ngày tháng năm sinh (ddmmyyyy)
- **Facebook:**
- **Microsoft Teams**
 - Account: Email Học viện (*@stu.ptit.edu.vn)
 - Mã nhóm:

1. Tổng quan về môn học Tin học cơ sở 2

❖ Dev C++

- <https://www.youtube.com/watch?v=dnwB6D7x9Cw>
- <https://www.youtube.com/watch?v=QtEd0oWAqlk>

❖ Visual Studio 2012, 2019

- <https://www.youtube.com/watch?v=hDcbHZZXhmc>
- <https://www.youtube.com/watch?v=w7yhp7IDpBQ>

❖ Code Block

❖

2. Tổng quan về ngôn ngữ lập trình

❖ Máy tính:

- Là thiết bị có khả năng thực hiện các phép tính và đưa ra các quyết định có tính logic.
- Xử lý dữ liệu dưới sự điều khiển của tập các chỉ thị (câu lệnh) - được gọi là chương trình máy tính.

❖ Phần cứng:

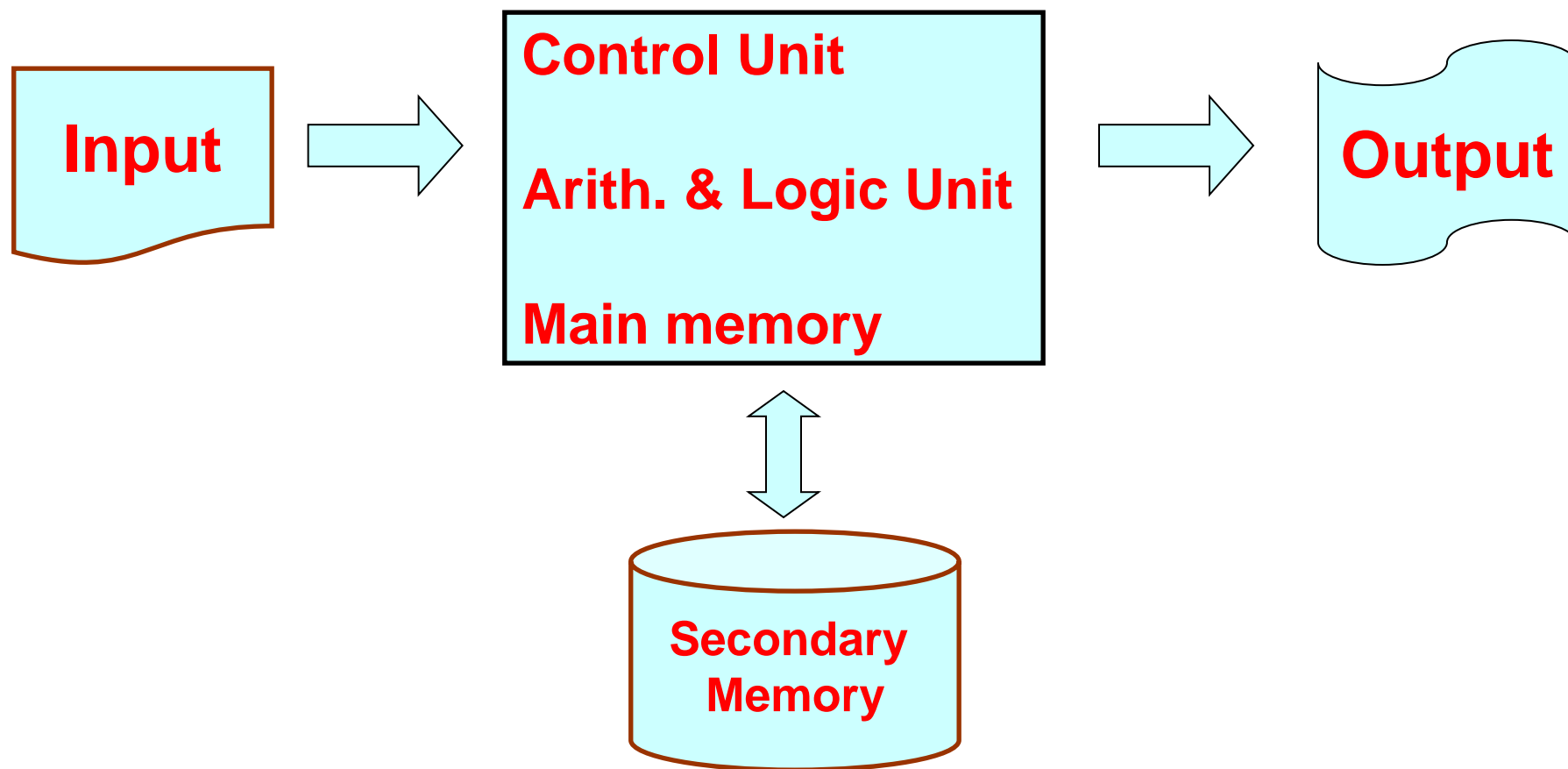
- Các thiết bị khác nhau cấu thành máy tính (các đơn vị xử lý, bộ nhớ, ổ đĩa, bàn phím,...)

❖ Phần mềm:

- Các chương trình chạy trên máy tính (hệ điều hành, chương trình ứng dụng).

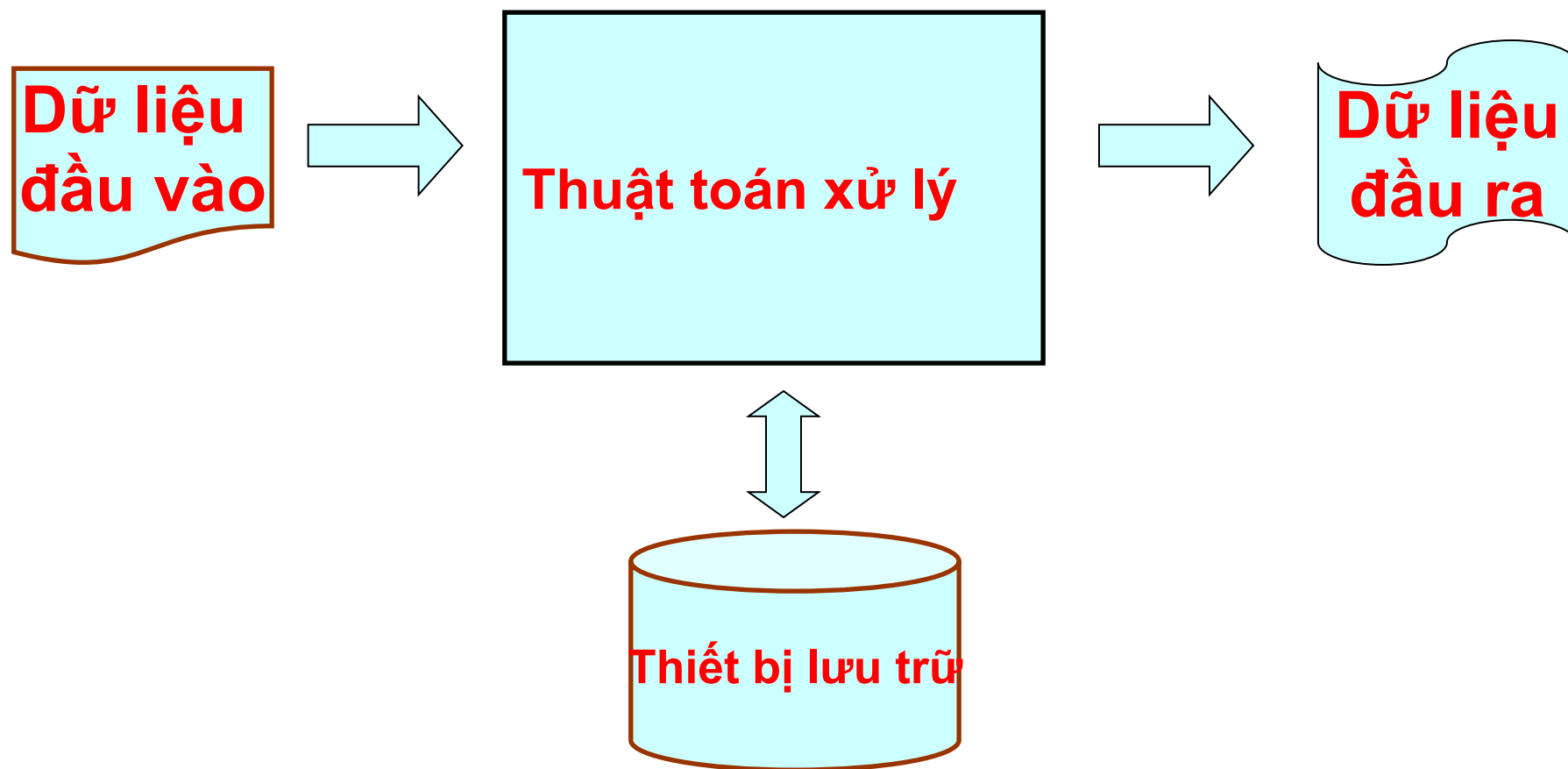
2. Tổng quan về ngôn ngữ lập trình

❖ Các thành phần của một hệ thống máy tính



2. Tổng quan về ngôn ngữ lập trình

❖ Các thành phần của chương trình



2. Tổng quan về ngôn ngữ lập trình

❖ Ngôn ngữ lập trình

Ngôn ngữ máy



Ngôn ngữ hợp ngữ



Ngôn ngữ cấp cao

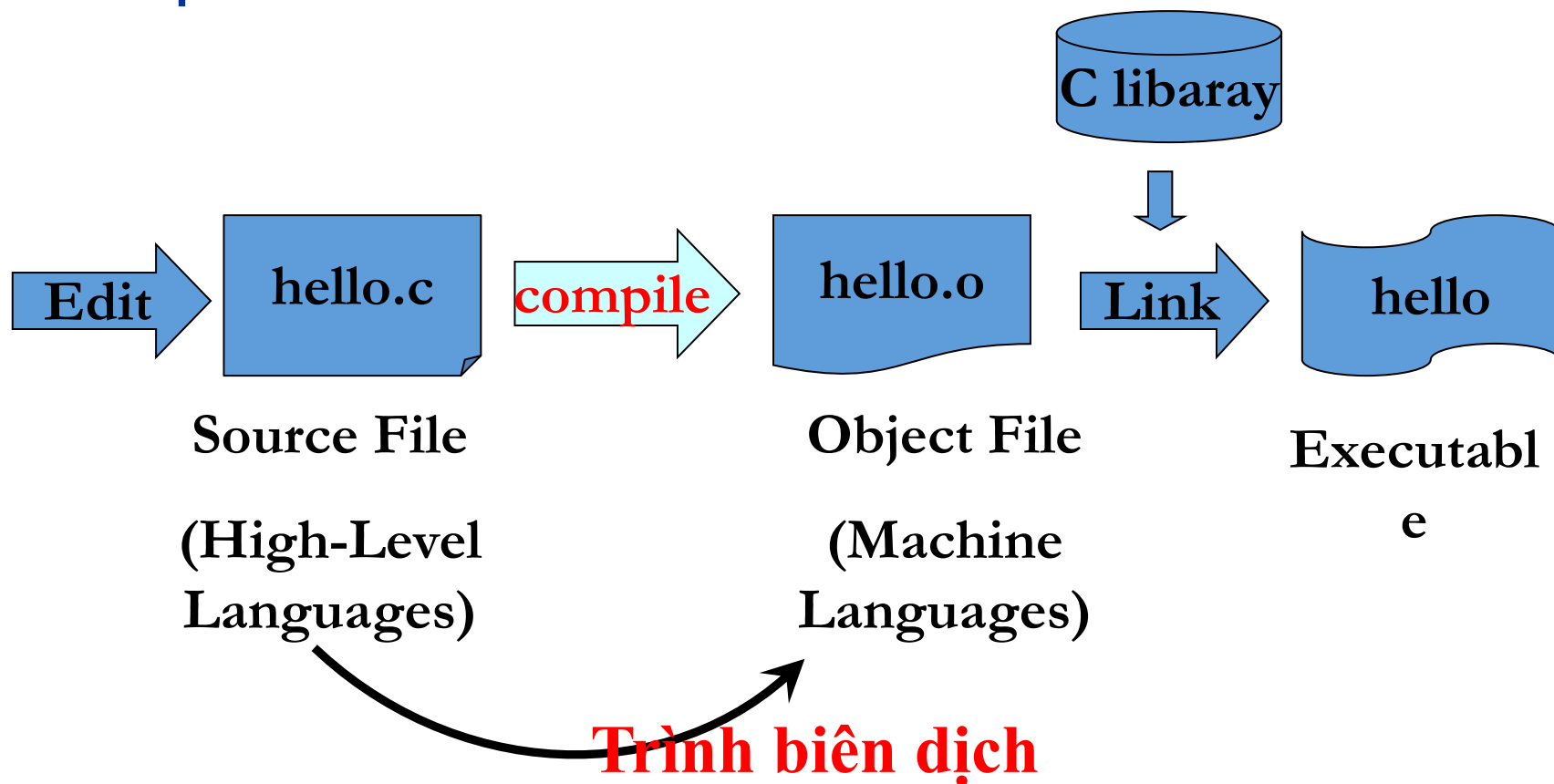
**+1300042774
+1400593419
+1200274027**

**MOV AX, A
MOV BX, B
ADD AX, BX**

A=A+B

2. Tổng quan về ngôn ngữ lập trình

❖ Trình dịch



2. Tổng quan về ngôn ngữ lập trình

❖ Trình dịch

- **Chương trình dịch** để chuyển đổi chương trình từ ngôn ngữ bậc cao (NN hợp ngữ) thành ngôn ngữ máy.
- **Trình biên dịch** (*compiler*): chuyển một chương trình ở NN cấp cao sang ngôn ngữ máy.
- **Trình thông dịch** (*interpreter*): phân tích và thực thi từng câu lệnh của chương trình NN cấp cao.

☞ C là ngôn ngữ cấp cao, dùng trình biên dịch.

2. Tổng quan về ngôn ngữ lập trình

❖ Lịch sử

- Ngôn ngữ C được thiết kế bởi Dennis Ritchie tại phòng thí nghiệm Bell (công ty AT&T - Mỹ) vào đầu những năm 1970.
- Chịu ảnh hưởng phát triển của một số ngôn ngữ:
 - ALGOL 60 (1960),
 - CPL (Cambridge, 1963),
 - BCPL (Martin Richard, 1967),
 - B (Ken Thompson, 1970)
- Được chuẩn hóa năm 1983 bởi ANSI (American National Standards Institute) và được gọi “ANSI C”

2. Tổng quan về ngôn ngữ lập trình

❖ C và C++

- C++ mở rộng từ C, được phát triển bởi Bjarne Stroustrup tại PTN. Bell – đầu 1980s.
- C++ có thêm nhiều khái niệm cơ bản mới, tạo cơ sở cho **lập trình hướng đối tượng** và một số đặc tính thuận lợi khác cho những dự án phát triển phần mềm.
- Một chương trình C tương thích trong cả C++ và C.

2. Tổng quan về ngôn ngữ lập trình

❖ Lập trình như thế nào?

- Phát biểu bài toán.
- Giải được bài toán.
- Viết **thuật toán** cho lời giải bài toán.
- Viết chương trình (*soạn thảo mã nguồn*) cho máy tính bằng một NNLT (C), lưu vào máy tính, gọi đây là **chương trình nguồn**.
- Gọi trình biên dịch: dịch chương trình nguồn sang mã máy; gọi trình liên kết để thực thi, đây là bước **chạy chương trình**.
 - Nếu trình biên dịch báo lỗi, đây là **lỗi cú pháp** \Rightarrow xem & sửa chương trình nguồn.
 - KQ mà MT giải được là không đúng yêu cầu, đây là **lỗi logic** \Rightarrow xem lại cách giải, thuật toán & chương trình.

4. Thuật toán

❖ Thuật toán-Algorithm là gì?

- Thuật toán là một dãy hữu hạn các bước, mỗi bước mô tả chính xác các phép toán hoặc hành động cần thực hiện, để giải quyết một vấn đề.
- 6 đặc trưng của thuật toán

2. Thuật toán

❖ Thuật toán-Algorithm là gì?

- 6 đặc trưng của thuật toán:
 - Input: Mỗi thuật toán có một số dữ liệu vào.
 - Output: Mỗi thuật toán có một số dữ liệu ra.
 - Tính xác định (Definiteness) Mỗi bước được mô tả chính xác, chỉ có một cách hiểu duy nhất và đủ đơn giản để có thể thực hiện được.
 - Tính dừng (Finiteness): Thuật toán phải dừng sau một số hữu hạn bước thực hiện
 - Tính hiệu quả (Effectiveness) Các phép toán trong các bước phải đủ đơn giản để có thể thực hiện được.
 - Tính tổng quát (Generalness) Thuật toán phải có tính tổng quát, có thể áp dụng cho một lớp đối tượng.

2. Thuật toán

❖ Biểu diễn thuật toán là gì?

- Ngôn ngữ tự nhiên (xem lại VD.PTB1)
- Mã giả (NNTN + NN LT)
- Sơ đồ khối: (chọn lựa & hành động).



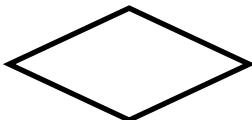
Khởi đầu & kết thúc thuật toán.



Biểu diễn thao tác xử lý.



Biểu diễn vào ra dữ liệu.



Thao tác chọn lựa theo điều kiện.



Chỉ đến bước thực thi tiếp theo.

2. Thuật toán

❖ Biểu diễn thuật toán là gì?

- Ưu điểm
 - Trình bày trực quan giải thuật
 - Độc lập với ngôn ngữ tự nhiên
 - Độc lập với ngôn ngữ lập trình
 - Bảo đảm khả năng lập trình
 - Cho phép dễ dàng kiểm tra giải thuật
- Nguyên tắc kiểm tra
 - Đi từ START theo bất cứ đường nào cũng phải đến một điểm dừng STOP
 - Không có sự quay vòng vĩnh viễn
 - Không có sự kết thúc lửng chừng

2. Thuật toán

❖ So sánh thuật toán và vẽ lưu đồ thuật toán:

1	2
Tính tổng của 2 số nguyên a,b	Nhập 2 số nguyên c,d. Tính tổng c+d và in kết quả ra màn hình.
Tính tổng của 2 số nguyên	Tính tổng của 2 số tự nhiên
Tính tổng của 2 số nguyên	Tính tổng của 2 số nguyên dương
Giải phương trình bậc 2	Giải phương trình bậc 2 với các hệ số khác 0
Tìm ước số chung lớn nhất của 2 số	Nhập 2 số nguyên dương a,b. Tìm ước số chung lớn nhất của 2 số và in kết quả ra màn hình
Tìm ước số chung lớn nhất của 2 số	Tìm ước số chung lớn nhất của 2 số tự nhiên
Kiểm tra số N có phải là số nguyên tố hay không?	Kiểm tra số N lớn hơn 1 có phải là số nguyên tố hay không?

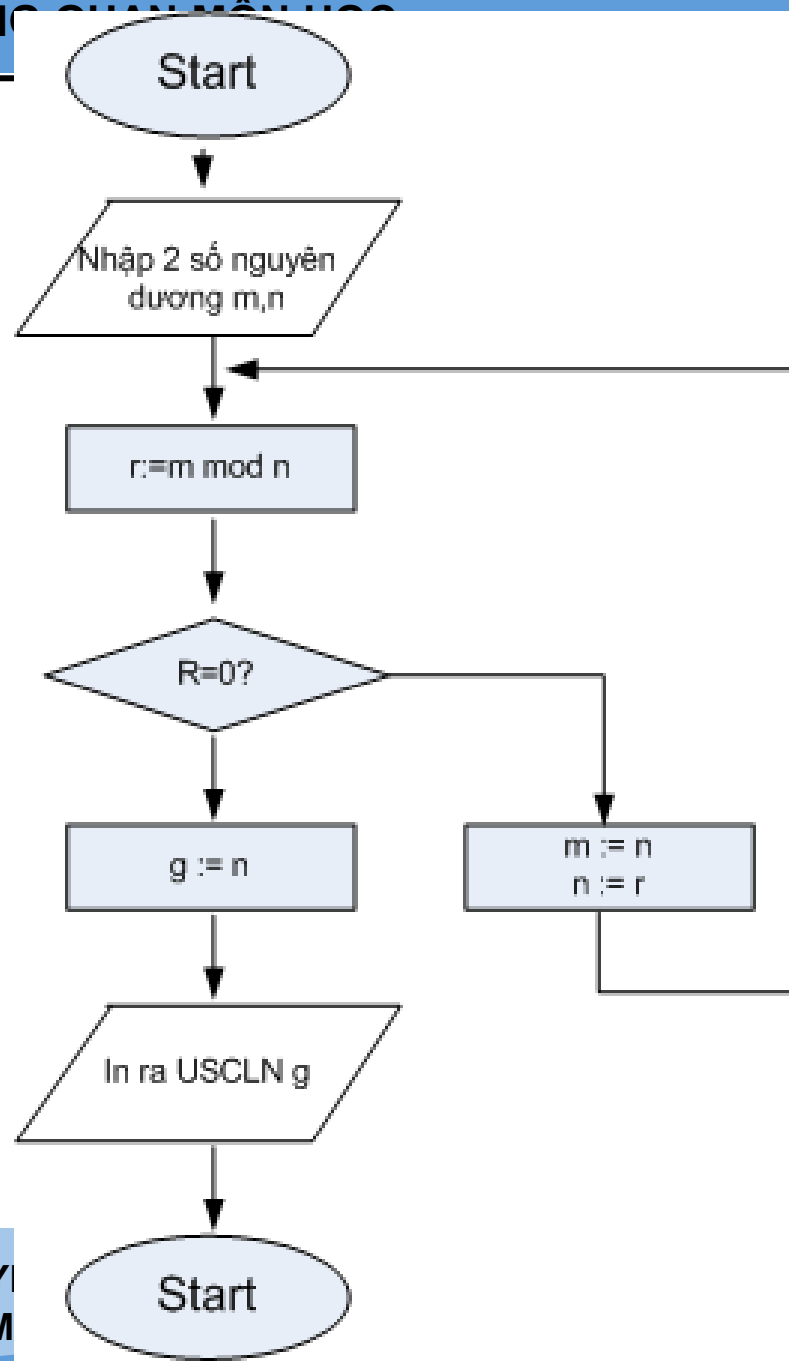
2. Thuật toán

❖ Ví dụ thuật toán:

- Thuật toán Euclid: Tìm ước số chung lớn nhất của hai số tự nhiên m, n .
- Input: m, n nguyên dương
- Output: g là ước số chung lớn nhất của m và n
- Phương pháp:
 - 1. $r = m \bmod n$
 - 2. Nếu $r=0$ thì $g:=n$, g là ƯSCLN
 - 3. Ngược lại ($r>0$) $m:=n$; $n:=r$ và quay lại bước 1.

2. Thuật toán

❖ Ví dụ thuật toán:



4. Bài tập về lưu đồ thuật toán

1. Giải phương trình bậc 1
2. Giải phương trình bậc 2
3. Kiểm tra 1 số có phải số nguyên tố?
4. Kiểm tra 2 số có nguyên tố cùng nhau hay không?
5. Tìm các số nguyên tố nhỏ hơn N.
6. Tính tổng của n số tự nhiên đầu tiên.
7. Tính tích của n số tự nhiên đầu tiên.

4. Bài tập suy nghĩ

1. Sắp xếp dãy số theo thứ tự tăng dần
2. Sắp xếp dãy số theo thứ tự giảm dần
3. In giá trị nhị phân của một số thập phân.
4. Đếm số chữ cái trong 1 chuỗi ký tự
5. Tần suất xuất hiện của một ký tự trong chuỗi ký tự
6. Kiểm tra 1 chuỗi ký tự có chứa 1 chuỗi ký tự khác không?