

Tài liệu chỉ xem được một số trang đầu. Vui lòng download file gốc để xem toàn bộ các trang

BỘ GIAO THÔNG VẬN TẢI
TRƯỜNG ĐẠI HỌC HÀNG HẢI
BỘ MÔN: TIN HỌC ĐẠI CƯƠNG
KHOA: CÔNG NGHỆ THÔNG TIN

BÀI GIẢNG

TIN HỌC ĐẠI CƯƠNG

TÊN HỌC PHẦN : **TIN HỌC ĐẠI CƯƠNG**
MÃ HỌC PHẦN : **17101**
TRÌNH ĐỘ ĐÀO TẠO : **ĐẠI HỌC CHÍNH QUY**
DÙNG CHO SV NGÀNH : **TẤT CẢ CÁC NGÀNH**

MỤC LỤC

STT	NỘI DUNG	TRANG
CHƯƠNG I	NHẬP MÔN TIN HỌC ĐẠI CƯƠNG	
1.1	Một vài nét về tin học và máy tính	4
1.2	Thông tin và xử lý thông tin	4
1.3	Thành phần của máy tính	4
1.4	Các hệ đếm	5
1.5	Phân loại máy tính	9
1.6	Các bộ phận chính của máy tính PC	10
CHƯƠNG II	TỔNG QUAN VỀ LẬP TRÌNH	
2.1	Vấn đề giải quyết bài toán bằng máy tính	12
2.2	Một số ví dụ	13
CHƯƠNG III	NGÔN NGỮ LẬP TRÌNH VISUAL BASIC	
3.1	Khởi động Visual Basic	20
3.2	Thoát khỏi Visual Basic	20
3.3	Tạo, mở và lưu một dự án (Project)	20
3.4	Các bước xây dựng chương trình	21
3.5	Một số đối tượng cơ bản trong Visual Basic	23
3.6	Một số ví dụ về thiết kế giao diện và lập bảng thuộc tính	29
3.7	Các kiểu dữ liệu cơ bản	30
CHƯƠNG IV	KHAI BÁO HẰNG BIẾN BIỂU THỨC CÂU LỆNH	
4.1	Các hằng trong Visual Basic	36
4.2	Biến	36
4.3	Biểu thức	37
4.4	Quy định về viết dòng lệnh trong Visual Basic	37
4.5	Cách viết dòng chú thích trong chương trình	37
4.6	Câu lệnh gán	37
4.7	Câu lệnh End	38
4.8	Lệnh in dữ liệu	38

4.9	Lệnh nhập dữ liệu	39
4.10	Chuyển đổi kiểu dữ liệu trong quá trình tính giá trị biểu thức	40
4.11	Cấu trúc điều kiện	41
4.12	Câu lệnh lựa chọn Select	43
4.13	Cấu trúc lặp	45
4.14	Một số ví dụ	48
CHƯƠNG V	HÀM VÀ THỦ TỤC	
5.1	Khái niệm chương trình con	51
5.2	Hàm và thủ tục	51
5.3	Truyền tham số cho chương trình con	53
5.4	Biến toàn cục, biến địa phương, khái niệm tầm tác dụng	56
CHƯƠNG VI	DỮ LIỆU CÓ CẤU TRÚC, KIỂU MẢNG	
6.1	Khái niệm về mảng	59
6.2	Khai báo mảng tĩnh	59
6.3	Khai báo mảng động (Dynamic Array)	61
6.4	Một số thuật toán về mảng	62

CHƯƠNG 1

NHẬP MÔN TIN HỌC ĐẠI CƯƠNG

1.1. Một vài nét về tin học và máy tính

Có rất nhiều khái niệm về Tin học, song có thể hiểu Tin học là ngành khoa học nghiên cứu các quá trình có tính chất thuật toán nhằm mô tả và biến đổi thông tin. Các quá trình này được nghiên cứu một cách hệ thống về mọi phương diện: Lý thuyết phân tích, thiết kế, tính hiệu quả, việc cài đặt và các ứng dụng.

1.2. Thông tin và xử lý thông tin

1.2.1. Thông tin và dữ liệu

Thông tin (Information): Những hiện tượng sự vật phản ánh về một sự kiện, một vấn đề nào đó trong thế giới khách quan. Dựa vào đó con người có thể hiểu biết và nhận thức được thế giới khách quan. Thông tin có thể ghi lại và truyền đi.

Dữ liệu (Data): Là dạng thông tin khi được lưu trữ.

1.2.2. Đơn vị đo thông tin

Đơn vị đo thông tin là bit. Đây chính là tin về hệ thống chỉ có hai trạng thái đồng khả năng: bằng 0 hoặc bằng 1 (Điều này rất phù hợp với các máy tính điện tử bởi trong một thời điểm, mạch điện chỉ có một trong hai trạng thái đóng hoặc mở tương ứng với hai giá trị 1 hoặc 0).

Đơn vị đo thông tin bao gồm: Bit, Byte, KiloByte, MegaByte, GigaByte.

Qui đổi giữa các đơn vị như sau:

Byte (B): 1 Byte = 8 bit (b)

KiloByte (KB): 1 KB = 1024 Byte

MegaByte (MB): 1 MB = 1024 KB

GigaByte (GB): 1 GB = 1024 MB

Trong đó: **b** là viết tắt của bit

B là viết tắt của Byte

1.3. Thành phần của máy tính

1.3.1. Phần cứng (Hardware)

Là các thành phần vật lý cấu tạo nên máy tính. Các thành phần vật lý ở đây bao gồm các thiết bị điện tử và cơ khí.

Ví dụ: về các phần cứng máy tính như màn hình, bàn phím, chuột, bộ vi xử lý...

1.3.2. Phần mềm (SoftWare)

Là tập hợp các chỉ thị cho máy tính làm việc. Nói cách khác, toàn bộ các chương trình chạy trên máy tính gọi là phần mềm máy tính. Sự ra đời của phần mềm khiến cho hiệu quả sử dụng phần cứng được nâng cao, rất nhiều công việc của con người được tự động hoá, vận hành nhanh chóng.

Ví dụ: Phần mềm soạn thảo văn bản, bảng tính, trình diễn, đồ hoạ...

1.4. Các hệ đếm

1.4.1. Hệ đếm thập phân (Decimal)

Hệ thập phân là hệ đếm dựa vào vị trí với cơ số 10. Hệ này dùng các số từ 0 đến 9 để biểu diễn. Giá trị của hệ thập phân được đánh giá bằng vị trí các con số.

Cách viết: 127 hoặc $(127)_{10}$

Khai triển một số hệ 10:

$$(123.78)_{10} = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 7 \times 10^{-1} + 8 \times 10^{-2}$$

1.4.2. Hệ đếm nhị phân (Binary)

Hệ nhị phân hay hệ đếm cơ số 2 chỉ có hai con số 0 và 1. Đó là hệ đếm theo vị trí. Giá trị của một số bất kỳ nào đó phụ thuộc vào vị trí của nó. Các vị trí có trọng số. Các vị trí của trọng số bằng bậc lũy thừa của cơ số 2.

Chấm cơ số được gọi là chấm nhị phân trong hệ đếm cơ số 2. Mỗi một con số nhị phân được gọi là một bit (Binary digit). Bit ngoài cùng bên trái là bit có trọng số lớn nhất (MSB) và bit ngoài cùng bên phải là bit có trọng số nhỏ nhất (LSB) như dưới đây:

$$\begin{array}{ccccccc} 2^3 & 2^2 & 2^1 & 2^0 & \bullet & 2^{-1} & 2^{-2} \\ \text{MSB} & 1 & 0 & 1 & 0 & \bullet & 1 & 1 & \text{LSB} \end{array}$$

Cách viết: $(1011)_2$

1.4.3. Hệ thập lục phân (HEXADECIMAL)

Các máy tính hiện đại thường dùng hệ đếm khác là hệ thập lục phân. Hệ thập lục phân là hệ đếm dựa vào vị trí với cơ số 16.

Hệ này dùng 10 chữ số từ 0 đến 9 và 6 ký tự từ A đến F để biểu diễn.

Hệ này thường dùng để viết gọn các số hệ nhị phân

Hệ này được biểu diễn như trong bảng sau:

Thập lục phân	Thập phân	Nhị phân
0	0	0000
1	1	0001
2	2	0010

3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

Cách viết : $(2BC)_H$

1.4.4. Chuyển đổi giữa các hệ đếm

* Chuyển từ hệ nhị phân sang hệ mười

Muốn đổi từ hệ nhị phân sang hệ mười chỉ cần tính các giá trị 2^i tương ứng với các chữ số khác không thứ i của hệ số nhị phân rồi cộng lại : (vị trí i tính từ phải qua trái)

Ví dụ 1: Ta muốn đổi một số hệ nhị phân $(101001)_2$ sang hệ mười, tiến hành như sau:

$$\begin{aligned}(101001)_2 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 32 + 0 + 8 + 0 + 0 + 1 = (41)_{10}\end{aligned}$$

Kết quả ta được số 41 ở hệ 10

Ví dụ 2: Ta muốn đổi một số hệ nhị phân $(101.001)_2$ sang hệ mười, tiến hành như sau:

$$\begin{aligned}(101.001)_2 &= 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 4 + 0 + 2 + 0 + 0 + 0.125 = (6.125)_{10}\end{aligned}$$

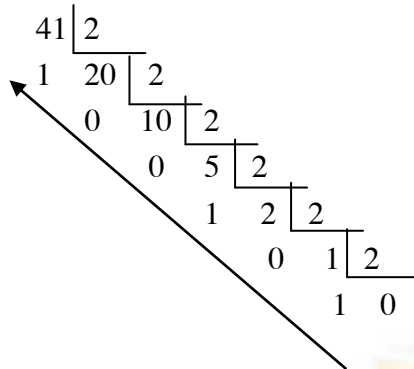
Chú ý: Trong các ví dụ trên quá trình biến đổi có thể đơn giản hoá bằng cách loại bỏ các con số không.

* Chuyển từ hệ thập phân sang hệ nhị phân:

Lấy phần nguyên của số cần đổi chia cho 2 và ghi nhớ phần dư, tiếp theo lấy thương của phép chia trước đó chia cho 2 và ghi nhớ phần dư. Cứ làm như vậy cho đến khi thương bằng 0 thì dừng lại và viết phần dư ngược từ dưới lên trên. Lấy phần lẻ thập phân của số cần đổi nhân với 2, tích nhận được gồm phần nguyên và phần lẻ nhị phân, lấy phần lẻ nhị phân của tích thu được nhân với 2. Cứ làm như vậy cho đến khi không còn

phần lẻ thập phân hoặc đến một độ chính xác nhất định nào đó thì dừng lại. Viết phần nguyên từ trên xuống dưới. Sau đó ghép kết quả của chúng lại với nhau.

Ví dụ 1: Đổi số $(41)_{10}$ từ hệ 10 sang hệ nhị phân:



Sau khi chia đến thương bằng 0 và viết ngược phần dư từ dưới lên trên ta được số $(101001)_2$ ở nhị phân.

Ví dụ 2: Biến đổi số thập phân $(41.625)_{10}$ thành số nhị phân:

Trước hết biến đổi phần nguyên thành số nhị phân theo ví dụ 1 được số $(101001)_2$

Sau đó biến đổi phần lẻ thành số nhị phân bằng cách: Lấy số cần đổi nhân với 2, tích nhận được gồm phần nguyên và phần lẻ nhị phân, lấy phần lẻ nhị phân của tích thu được nhân với 2. Cứ làm như vậy cho đến khi được tích chẵn bằng 1. Chọn riêng các phần nguyên (phần trước dấu phẩy) của các tích thu được và sắp xếp lại sẽ được các chữ số sau dấu phẩy cần tìm.

$$\begin{array}{r}
 0.625 \times 2 = 1,250 \\
 1.250 \times 2 = 0,500 \\
 0.5 \times 2 = 1,000
 \end{array}$$

phần lẻ $(0.625)_{10}$ được chuyển thành $(101)_2$ hệ 2

Vậy số $(41.625)_{10}$ ở hệ 10 sang hệ nhị phân là $(101001.101)_2$

***Chuyển từ hệ thập lục phân sang hệ thập phân:**

Các số thập lục phân có thể được biến đổi thành thập phân bằng cách tính tổng của các con số nhân với giá trị vị trí của nó.

Ví dụ 1: Đổi số $(5B)_{16}$ thành số thập phân

$$\begin{aligned}
 (5B)_{16} &= 5 \times 16^1 + B \times 16^0 \\
 &= 5 \times 16^1 + 11 \times 16^0 \\
 &= 80 + 11 = (91)_{10}
 \end{aligned}$$

Kết quả là số $(91)_{10}$ hệ 10

Ví dụ 2: Đổi số $(2AF)_{16}$ thành số thập phân

$$\begin{aligned}(2AF)_{16} &= 2 \times 16^2 + A \times 16^1 + F \times 16^0 \\ &= 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 \\ &= 2 \times 256 + 10 \times 16 + 15 \times 1 \\ &= 512 + 160 + 15 = (687)_{10}\end{aligned}$$

Kết quả là số $(687)_{10}$ hệ 10

***Chuyển từ hệ thập phân sang hệ thập lục phân:**

Lấy số thập phân cần đổi chia cho 16 và ghi nhớ phần dư, tiếp theo lấy thương của phép chia trước đó chia cho 16 và ghi nhớ phần dư. Cứ làm như vậy cho đến khi thương bằng 0 thì dừng lại và viết phần dư ngược từ dưới lên trên.

Ví dụ: Đổi số $(93)_{10}$ thành hệ thập lục phân:

	Thương số	+	số dư	
$\frac{93}{16}$	=	5	+	13 tương ứng là D
$\frac{5}{16}$	=	0	+	5

Vậy kết quả là số $(93)_{10}$ đổi thành số $(5D)_{16}$ hệ thập lục phân

***Chuyển từ hệ thập lục phân sang hệ nhị phân:**

Thực chất các số thập lục phân cũng chỉ là cách biểu diễn các số nhị phân thuận lợi hơn. Muốn đổi từ số thập lục phân sang số hệ nhị phân ta chỉ cần thay thế từ con số thập lục phân bằng 4 bit nhị phân tương ứng:

Ví dụ: Đổi số $(C5)_{16}$ thành hệ nhị phân

C	5
↓	↓
1100	0101

Kết quả số $(C5)_{16}$ thành số $(11000101)_2$

***Chuyển từ hệ nhị phân sang hệ thập lục phân:**

Để biến đổi một số nhị phân thành số thập lục phân tương đương chỉ cần nhóm 4 bit tính từ phải qua trái sau đó chuyển từng nhóm 4 bit thành hệ thập lục phân và ghép lại với nhau sẽ được số hệ thập lục phân.

Ví dụ: Đổi số $(01011111)_2$ thành số thập lục phân

0101	1111
↓	↓
5	F

Kết quả số $(01011111)_2$ thành số $(5F)_{16}$

1.5. Phân loại máy tính

Theo góc độ người sử dụng thì máy tính được phân làm bốn loại sau:

* Máy tính lớn (Mainframe)

Máy tính lớn là máy có công suất cao, tốc độ xử lý nhanh và rất lớn thường đóng vai trò trong các hệ tin học phân tán có qui mô lớn.

Các máy tính lớn là các cỗ máy kích thước lớn, mạnh và rất đắt tiền, được sử dụng trong hầu hết các cơ quan, tổ chức lớn (ví dụ như các ngân hàng, các cơ quan trọng yếu của Chính phủ, các công ty đa quốc gia...).

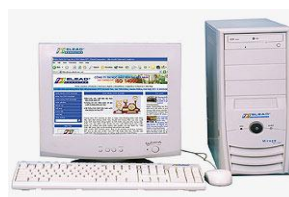
Cần chú ý phân biệt máy tính lớn với máy tính cá nhân có kích thước lớn.

* Máy tính PC (Personal Computer-PC)

Máy tính điện tử có nguồn gốc ra đời từ khá sớm và có rất nhiều chủng loại. Tuy nhiên chiếc máy tính PC theo mô hình của IBM đầu tiên được ra đời vào năm 1981. Từ đó trở đi, tất cả các máy tính PC được sản xuất, qua nhiều lần mở rộng đều tương thích với thiết kế ban đầu. Thuật ngữ máy tính tương thích liên quan đến các máy tính được sản xuất bởi các công ty khác không phải IBM nhưng vẫn tương thích với các đặc điểm kỹ thuật truyền thống.



IBM – PC
(1981)



IBM – PC
Tương thích

* Máy Mac (Apple MAC)

Máy Mac là một máy tính, nhưng không phải là một máy PC theo mô hình IBM. Máy Mac là dòng máy tính do hãng Apple sản xuất và chạy Hệ điều hành Macintosh của Apple có giao diện đồ họa GUI và tương thích với nó là các phiên bản đặc biệt của các chương trình ứng dụng (như bộ xử lý văn bản và các bảng tính). Thậm chí để có thể kết nối với một Mac, người ta cần phải lựa chọn thêm các thiết bị phần cứng mở rộng.



Thời kỳ đầu, điều giúp cho việc phân biệt máy Mac và PC chính là việc MAC sử dụng giao diện đồ họa cho người dùng (GUI – Graphic User Interface), hay nói cách khác, chính là cách mà ta có thể dùng chuột để điều khiển máy tính. Khi PC mới xuất hiện, người dùng phải là một nhà chuyên môn mới có thể sử dụng được chiếc PC đó. Tuy nhiên, gần đây sự khác biệt

cơ bản giữa máy MAC và máy PC đã không còn rõ rệt như trước, nhất là khi Microsoft đã sử dụng Graphic User Interface như là công cụ trợ giúp người dùng máy tính PC.

(Chú ý: Ý tưởng sử dụng các giao diện đồ họa và tương tác người – máy thông qua thao tác chuột là do Apple đề xuất, không phải do Microsoft)

* Máy tính xách tay (Laptop)

Laptop, tiếng Anh có nghĩa là đặt trong lòng, phân biệt với máy tính đặt trên mặt bàn là desktop. Ở Việt Nam, ta hay gọi Laptop là máy tính xách tay bởi đây là một loại máy tính nhỏ có thể mang đi theo, có thể chạy bằng pin. Ngoài ra, người ta còn sử dụng tên gọi “Notebooks” (quyển vở/quyển sổ) để chỉ các máy tính xách tay có kích thước nhỏ hơn.



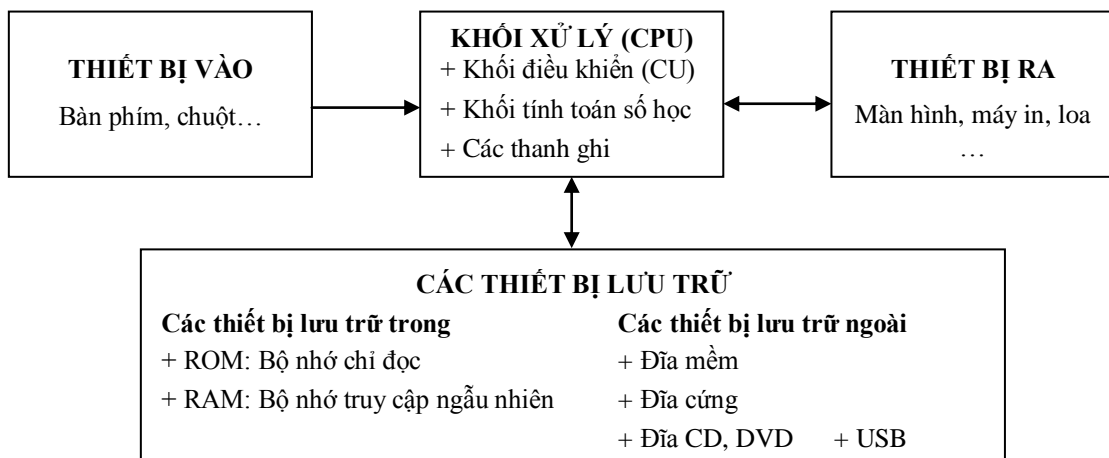
Cả hai dòng máy tính PC và MAC đều có những sản phẩm máy tính xách tay cho người dùng của mình. Do đặc điểm nhỏ gọn và năng lực xử lý khá tốt nên máy tính xách tay rất được ưa chuộng, đặc biệt là những người hay phải di chuyển hoặc phải thuyết trình trước đám đông. Những người thường xuyên sử dụng máy tính xách tay có thể kể đến những người làm công tác kinh doanh (chuyên đi thuyết trình sản phẩm), những người làm trong ngành giáo dục (giảng bài bằng máy tính xách tay kết nối máy chiếu)...

Chú ý: Thực chất máy tính xách tay cũng là dòng máy PC hay máy Mac là dòng máy PC nhưng do hãng Apple sản xuất.

1.6. Các bộ phận chính của máy tính PC

1.6.1. Sơ đồ các khối chức năng và bộ phận chính trong máy tính

Về cơ bản, thành phần chính của một máy tính bao gồm các khối như: Khối thiết bị vào/ Khối xử lý/ Khối thiết bị ra/ Khối thiết bị lưu trữ.



1.6.2. Khối xử lý trung tâm (Central Processing Unit – CPU)

Khối xử lý trung tâm, hay còn gọi là một bộ vi xử lý hoặc con chip, là bộ não của máy tính. Công việc chính của khối xử lý trung tâm là tính toán và điều khiển mọi hoạt động trong máy tính.



1.6.3. Bộ nhớ trong (Internal Memory)

Bộ nhớ trong máy tính dùng để chứa các lệnh và dữ liệu phục vụ cho quá trình thực hiện các chương trình.

Bộ nhớ ROM (Read Only Memory): Dữ liệu được đưa vào từ nhà sản xuất, thông tin trong ROM sẽ không bị mất khi tắt máy. Bộ nhớ ROM không thể nâng cấp hoặc thay thế.

Bộ nhớ RAM (Random Access Memory): Là nơi lưu trữ tạm thời dữ liệu của người sử dụng trong quá trình xử lý. Dữ liệu trên RAM sẽ bị mất khi tắt máy. Bộ nhớ Ram có thể được nâng cấp hoặc thay thế.

1.6.4. Bộ nhớ ngoài (External Memory)

Bộ nhớ ngoài hay các thiết bị lưu trữ ngoài bao gồm đĩa cứng, đĩa mềm, đĩa CD, DVD, đĩa Zip...

1.6.5. Các thiết bị vào (Input device)

Các thiết bị vào cho phép thông tin hay dữ liệu được nhập vào máy tính, ví dụ như bàn phím, chuột, máy quét...

1.6.6. Các thiết bị ra (Output device)

Các thiết bị ra cho phép thông tin có thể được xuất ra từ máy tính, ví dụ như máy in, màn hình, loa...

1.6.7. Các thiết bị ngoại vi (Peripheral device)

Thiết bị ngoại vi là thiết bị mà nó có thể gắn/cắm vào máy tính. Như vậy toàn bộ các thiết bị như máy quét, máy in, bàn phím, chuột, loa... đều là các thiết bị ngoại vi.

CHƯƠNG 2

TỔNG QUAN VỀ LẬP TRÌNH

2.1. Vấn đề giải quyết bài toán bằng máy tính

2.1.1. Lập trình (Programming)

Lập trình dùng để chỉ thao tác của con người nhằm kiến tạo, xây dựng một kịch bản bằng các thuật toán để giải quyết một bài toán Tin học trên máy tính thông qua các ngôn ngữ lập trình.

2.1.2. Ngôn ngữ lập trình (Programming language)

Ngôn ngữ lập trình là một tập con của ngôn ngữ máy tính. Đây là một dạng ngôn ngữ được chuẩn hóa (đối lập với ngôn ngữ tự nhiên). Nó là một hệ thống được ký hiệu hóa để miêu tả những tính toán (qua máy tính) trong một dạng mà cả con người và máy đều có thể đọc và hiểu. Ngôn ngữ lập trình cũng chính là một chương trình, nhưng có thể được dùng để tạo nên các chương trình khác. Một chương trình máy tính được viết bằng một ngôn ngữ lập trình thì những chỉ thị (của riêng ngôn ngữ ấy) góp phần tạo nên chương trình được gọi là mã nguồn của chương trình ấy.

2.1.3. Bài toán Tin học

Bài toán tin học không dùng để chỉ một bài toán cụ thể, mà dùng để chỉ một lớp các bài toán cụ thể thuộc cùng một loại. Một bài toán tin học được cấu tạo bởi hai yếu tố sau:

- + Thông tin đầu vào (Input): Là những thông tin bài toán đã cho
- + Thông tin đầu ra (Output): Là các thông tin cần tìm hoặc câu trả lời cần thiết.

Ví dụ: Giải bài toán tính diện tích tam giác bằng công thức $s=a*h/2$

- + Thông tin đầu vào: Là cạnh đáy a và đường cao h
- + Thông tin đầu ra: Là diện tích tam giác S hoặc dòng thông báo dữ liệu không hợp lệ.

2.1.4. Thuật toán

Thuật toán là tập hợp hữu hạn các bước, các phép toán cơ bản nhất được sắp xếp theo một trình tự nhất định để từ thông tin đầu vào của bài toán sau một tập hữu hạn các bước đó sẽ đạt được kết quả ở đầu ra như mong muốn.

Ví dụ: Trở lại bài toán tính diện tích tam giác, thuật toán như sau:

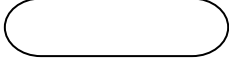
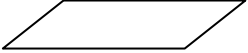
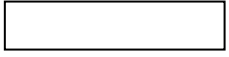
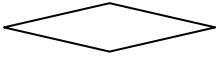
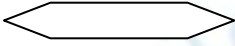

- + **Bước 1:** Cho giá trị của cạnh đáy a và đường cao h
- + **Bước 2:** Kiểm tra
 - Nếu $a>0$ và $h>0$ thì tính diện tích theo công thức $s=a*h/2$ sau đó xuống bước 3.
 - Ngược lại thông báo dữ liệu không hợp lệ và quay về bước 1
- + **Bước 3:** In diện tích S

2.1.5. Biểu diễn thuật toán qua sơ đồ khối (A.Flow Chart)

Biểu diễn thuật toán: Là sơ đồ dùng các ký hiệu và các đường nối với nhau để diễn tả các quan hệ logic và trình tự thực hiện các phép toán của một chương trình máy tính.

Để thể hiện trực quan về thuật toán người ta dùng sơ đồ khối. Qua sơ đồ khối người lập trình có thể quan sát các bước của thuật toán cũng như dòng thông tin hình thành và biến đổi trong quá trình thuật toán làm việc.

Việc thể hiện thuật toán người ta quy ước dùng các khối sau để thể hiện.

-  - Thể hiện sự bắt đầu và kết thúc chương trình
-  - Thể hiện việc nhập, xuất dữ liệu
-  - Chứa các công thức tính toán
-  - Chứa các biểu thức kiểm tra điều kiện và rẽ nhánh chương trình
-  - Gọi chương trình con
-  - Chỉ ra hướng đi của thuật toán và kết nối giữa các hình

2.2. Một số ví dụ

Ví dụ 1:

Lập thuật toán và vẽ sơ đồ thuật toán để giải và biện luận phương trình $ax + b = 0$

Bài giải:

Xây dựng thuật toán:

- Dữ liệu đầu vào là giá trị hai số a và b (trong lập trình thường gọi tắt là nhập giá trị a và b)

- Kiểm tra điều kiện của a :

Nếu $a = 0$ thì

+ Nếu $b=0$ thì

Phương trình vô số nghiệm

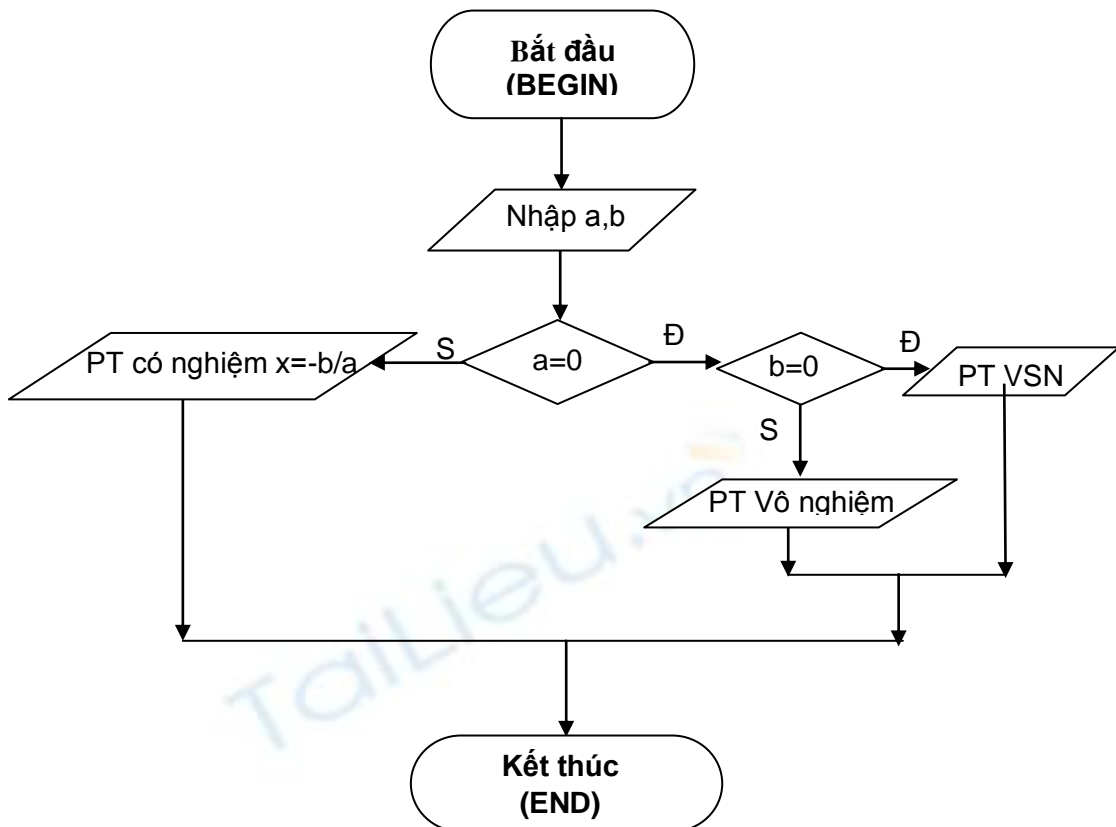
Ngược lại

Phương trình vô nghiệm

Ngược lại

Phương trình có nghiệm $x=-b/a$

Sơ đồ thuật toán:



Ví dụ 2:

Cho ba số bất kỳ a,b,c hãy tìm giá trị lớn nhất trong ba số này

Bài giải:

Xây dựng thuật toán:

- Nhập giá trị cho ba biến a,b,c
- Coi một trong ba số là số lớn nhất (giả sử số a) đưa giá trị của a vào một biến max

Mô tả trên được viết như sau $max=a$

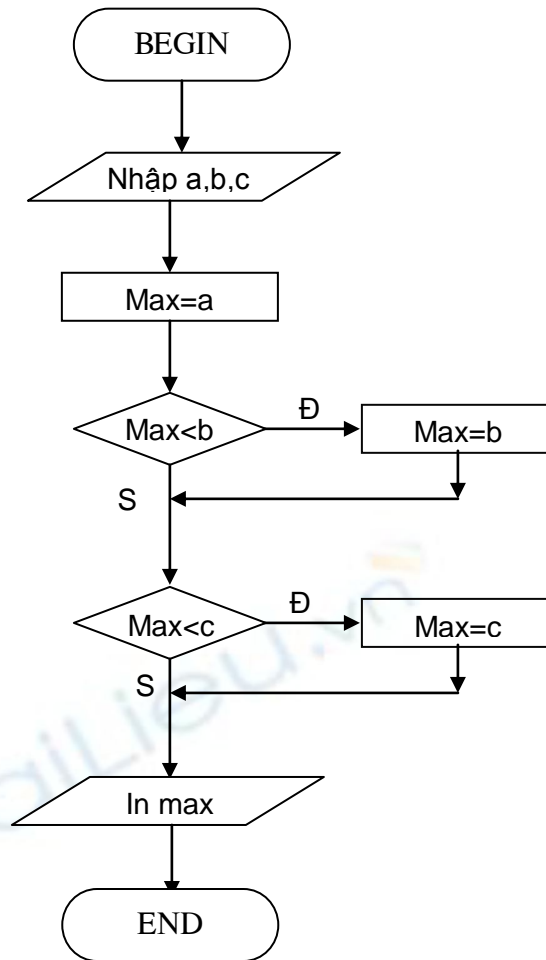
- Kiểm tra giá trị của biến max với từng số còn lại

Nếu $max < b$ thì $max=b$

Nếu $max < c$ thì $max=c$

- In giá trị lớn nhất trong max ra màn hình

Sơ đồ thuật toán:



Ví dụ 3: Hãy tính Tổng $S=1+1/2+1/3+\dots+1/(n-1)+1/n$ với n là một số tự nhiên.

Bài giải:

Xây dựng thuật toán:

Bước 1: Nhập giá trị cho n

Bước 2: Kiểm tra giá trị của $n>0$

+ Nếu sai: Trở về bước 1

+ Nếu đúng: Chuyển sang bước 3

Bước 3: Cho $S=0$

Khởi tạo biến chạy i và cho $i=1$

Bước 4: Kiểm tra điều kiện của i

- Nếu $i \leq n$ thì:

+ Tính $S=S+1/i$

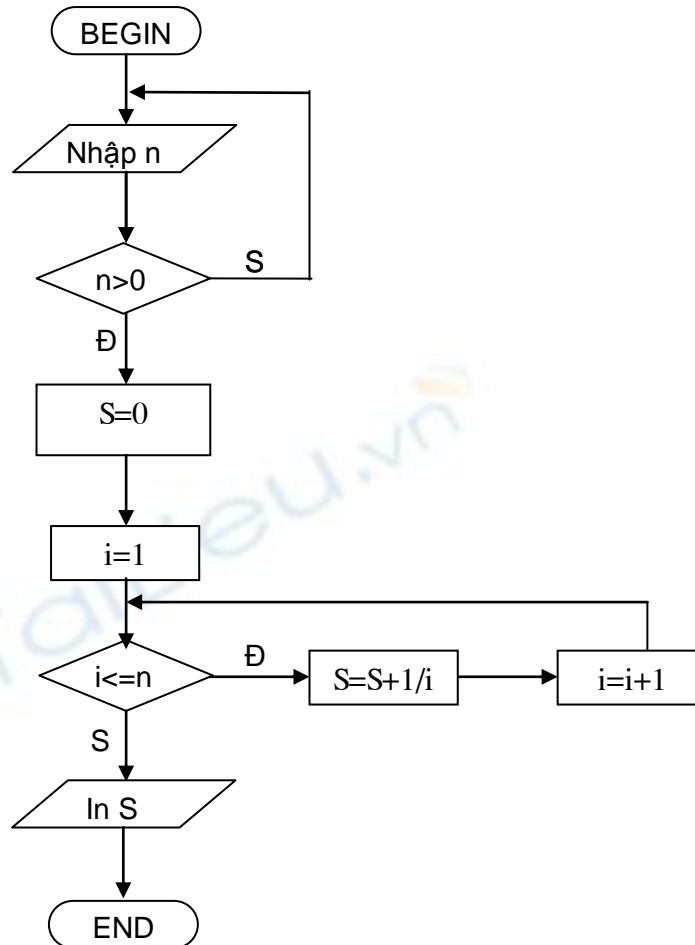
+ Tăng i : $i=i+1$

+ Quay trở về đầu bước 4

- Nếu $i > n$: Chuyển sang bước 5

Bước 5: In giá trị của S

Sơ đồ thuật toán:



Ví dụ 4:

Nhập vào một dãy số nguyên gồm n phần tử. Sau đó tính trung bình cộng các số chia hết cho 3.

Phân tích bài toán:

Dữ liệu đầu vào:

- Số phần tử của dãy n (n là số nguyên dương)
- Dãy số có n phần tử bất kỳ

Dữ liệu đầu ra: - Trung bình cộng của các số chia hết cho 3

Để tính được trung bình cộng ta cần phải tính Tổng và số lượng các số chia hết cho 3. Theo dữ liệu đầu vào của bài toán ta có thể giải quyết theo hai hướng sau:

Hướng 1: Lưu trữ dãy số bằng một ô nhớ	Hướng 2: Lưu trữ dãy số bằng n ô nhớ
<p>Bước 1: - Xác định số phần tử của dãy, nhập giá trị cho n.</p> <p>Bước 2:</p> <ul style="list-style-type: none"> - Kiểm tra điều kiện n <ul style="list-style-type: none"> + Nếu $n \leq 0$: Quay về bước 1 + Nếu $n > 0$: Chuyển sang bước 3 <p>Bước 3:</p> <p>Để tính TBC trước hết ta phải tính:</p> <ul style="list-style-type: none"> + Tổng các số thoả mãn, gọi tổng đó là T, khởi tạo giá trị ban đầu $T=0$ + Số lượng các phần tử thoả mãn, gọi là d, khởi tạo giá trị ban đầu $d=0$ - Khởi tạo biến chạy $i=1$ (coi như khi $i=1$ ta tiến hành nhập giá trị cho phần tử thứ nhất) <p>Bước 4:</p> <p>Kiểm tra điều kiện của i.</p> <ul style="list-style-type: none"> - Nếu $i \leq n$ <ul style="list-style-type: none"> + Nhập giá trị cho a + Kiểm tra: Nếu a chia hết cho 3 thì $T=T+a$ $d=d+1$ + Tăng biến i lên một đơn vị: $i=i+1$ + Quay về bước 4 - Nếu $i > n$: Chuyển sang bước 5 <p>Bước 5:</p> <ul style="list-style-type: none"> - Sau khi tính được T và d ta cần kiểm tra lại d vì có thể xảy ra trường hợp trong n phần tử không có phần tử nào chia hết cho 3 (tương ứng với $d=0$) thì không thể tính được TBC. - Nếu $d=0$ thì <ul style="list-style-type: none"> Dãy số không có phần tử thoả mãn 	<p>Bước 1: - Xác định số phần tử của dãy, nhập giá trị cho n.</p> <p>Bước 2:</p> <ul style="list-style-type: none"> - Kiểm tra điều kiện n <ul style="list-style-type: none"> + Nếu $n \leq 0$: Quay về bước 1 + Nếu $n > 0$: Chuyển sang bước 3 <p>Bước 3:</p> <p>Để tính TBC trước hết ta phải tính:</p> <ul style="list-style-type: none"> + Tổng các số thoả mãn, gọi tổng đó là T, khởi tạo giá trị ban đầu $T=0$ + Số lượng các phần tử thoả mãn, gọi là d, khởi tạo giá trị ban đầu $d=0$ - Khởi tạo biến chạy $i=1$ (coi như khi $i=1$ ta tiến hành nhập giá trị cho phần tử thứ nhất) <p>Bước 4:</p> <p>‘Nhập giá trị liên một lúc cho n phần tử chứa vào n ô nhớ a_1, a_2, \dots, a_n</p> <p>Kiểm tra điều kiện của i.</p> <ul style="list-style-type: none"> - Nếu $i \leq n$ <ul style="list-style-type: none"> + Nhập giá trị cho phần tử a_i + Tăng biến i lên một đơn vị: $i=i+1$ + Quay về bước 4 - Nếu $i > n$: Chuyển sang bước 5 <p>Bước 5:</p> <p>‘ Tính toán TBC</p> <ul style="list-style-type: none"> - Khởi tạo biến chạy $i=1$ <p>Bước 6:</p> <p>Kiểm tra điều kiện của i.</p> <ul style="list-style-type: none"> - Nếu $i \leq n$ <ul style="list-style-type: none"> + Kiểm tra: Nếu a chia hết cho 3 thì $T=T+a$ $d=d+1$ + Tăng biến i lên một đơn vị: $i=i+1$

điều kiện chia hết cho 3

ngược lại

Tính trung bình cộng: $TBC = T/d$

In giá trị của TBC ra

Bước 6: Kết thúc

+ Quay về bước 6

- Nếu $i > n$: Chuyển sang bước 7

Bước 7:

- Sau khi tính được T và d ta cần kiểm tra lại d vì có thể xảy ra trường hợp trong n phần tử không có phần tử nào chia hết cho 3 (tương ứng với $d=0$) thì không thể tính được TBC.

- Nếu $d=0$ thì

Dãy số không có phần tử thoả mãn điều kiện chia hết cho 3

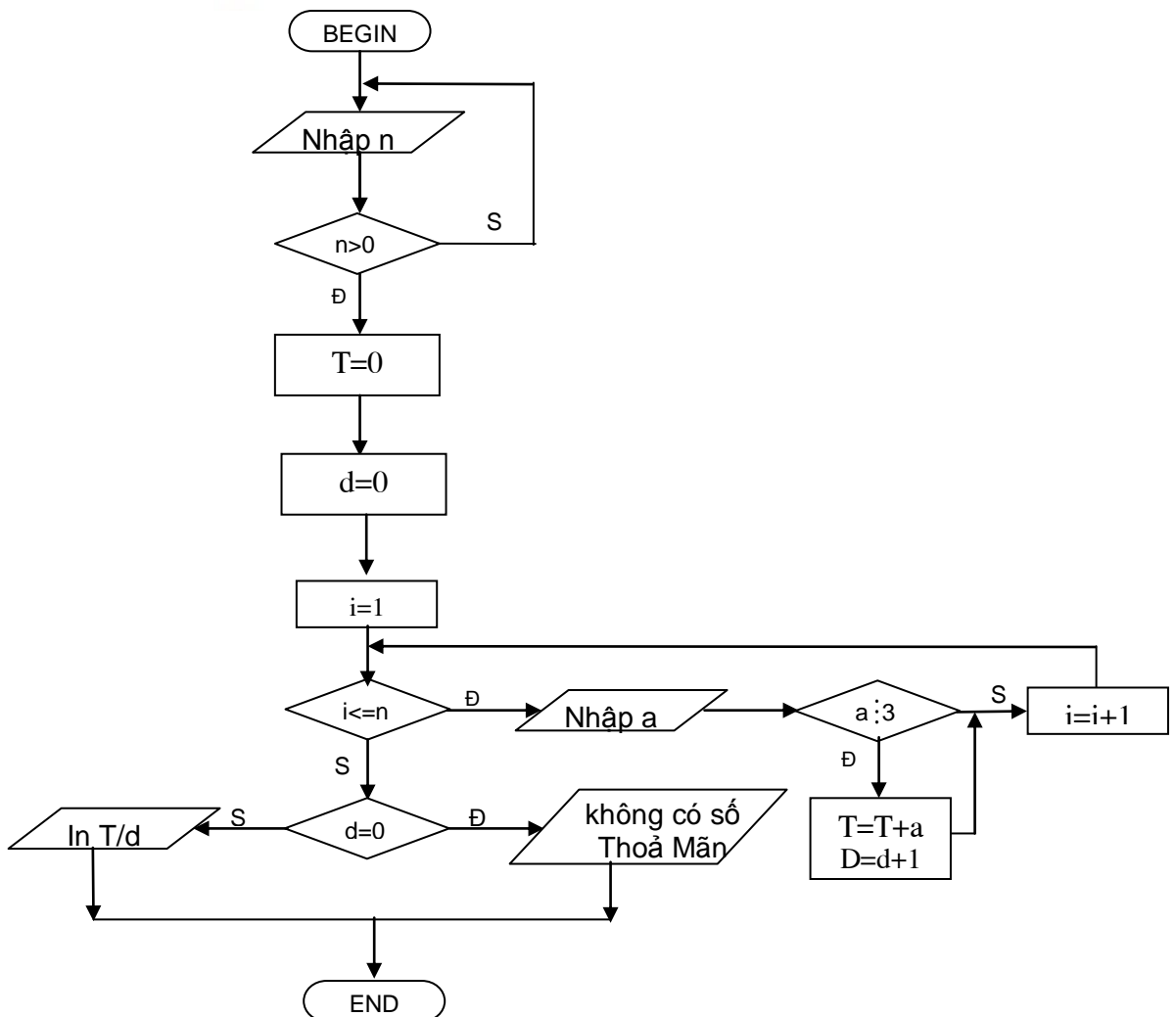
ngược lại

Tính trung bình cộng: $TBC = T/d$

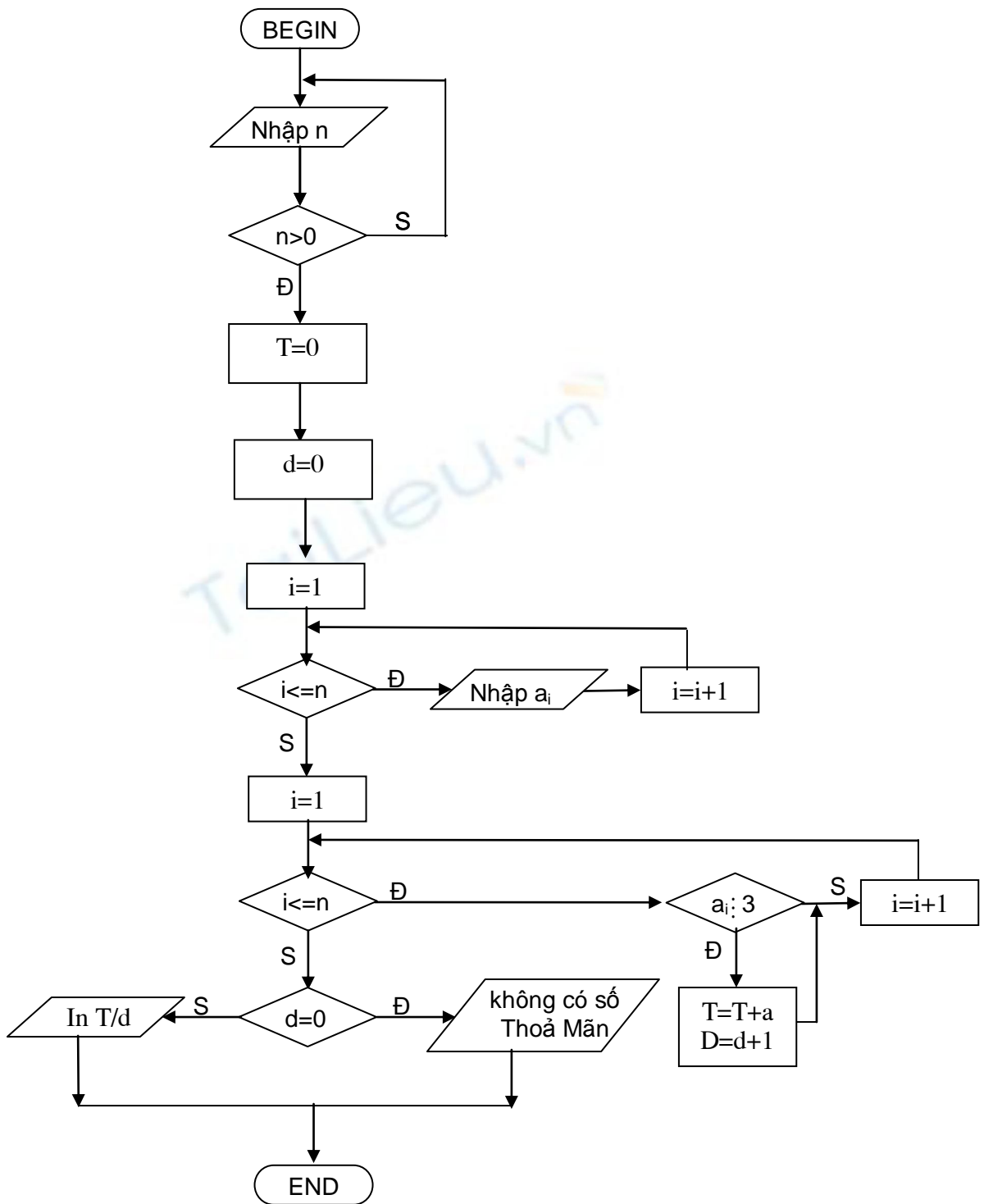
In giá trị của TBC ra

Bước 8: Kết thúc

Sơ đồ thuật toán theo hướng 1:



Sơ đồ thuật toán theo hướng 2:



CHƯƠNG 3

NGÔN NGỮ LẬP TRÌNH VISUAL BASIC

3.1. Khởi động Visual Basic

Kích chuột vào nút **Start\Programs\MiscoSoft Visual Studio 6.0\MicroSoft Visual Basic 6.0**

3.2. Thoát khỏi Visual Basic

Để thoát khỏi Visual Basic ta làm theo cách sau:

Vào menu **File\Exit**. Khi thoát khỏi Visual Basic nếu ta chưa ghi lại chương trình và các thành phần liên quan đến chương trình Visual Basic sẽ nhắc nhở chúng ta ghi lại.

3.3. Tạo, mở và lưu một dự án (Project)

Bước 1: Trên thư mục gốc của ổ đĩa C tạo thư mục có tên là VIDU

Lý do cần phải tạo riêng ra một thư mục cho chương trình vì một chương trình viết bằng Visual Basic bao gồm nhiều thành phần như: Các màn hình giao diện (Form), các đơn thể chương trình (Modul), các báo cáo (Report)... mỗi một thành phần sẽ được tạo các file tương ứng trên ổ đĩa; chính vì vậy để quản lý chương trình cho tốt phong cách lập trình tốt là hãy tạo riêng cho mỗi chương trình một thư mục riêng.

Bước 2: Khởi động VB khi đó hộp hội thoại **New Project** xuất hiện:



Trong hộp hội thoại này chọn tab New sau đó chọn loại Project muốn tạo, ở đây ta sẽ chọn StandardEXE. Hai tab còn lại có ý nghĩa sau:

- Existing: Mở Project đã có
- Recent: Mở Project đã làm việc trong thời gian gần đây nhất.