



Chương 1

CÁC KHÁI NIỆM CƠ BẢN

VỀ LẬP TRÌNH

Khoa Hệ thống thông tin quản lý

Hà Nội – 2015

Nội dung

1

Các khái niệm cơ bản

2

Các bước xây dựng chương trình

3

Thuật toán và chương trình

4

Giới thiệu ngôn ngữ lập trình C

1. Các khái niệm cơ bản

□ Lập trình (programming)

- Nghệ thuật cài đặt một hoặc nhiều thuật toán trừu tượng có liên quan với nhau bằng một ngôn ngữ lập trình để tạo ra một chương trình máy tính.

□ Bài toán

- Là việc nào đó ta muốn máy thực hiện để từ thông tin đưa vào (INPUT) tìm được thông tin ra (OUTPUT)
- Ví dụ: Giải phương trình bậc nhất $ax + b = 0$
 - INPUT: **a, b thuộc R**
 - OUTPUT: nghiệm của phương trình $ax + b = 0$

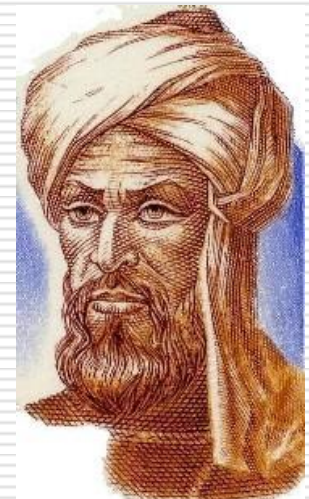
1. Các khái niệm cơ bản

□ Thuật toán (Algorithm)

- Thuật toán để giải một bài toán là một **dãy hữu hạn** các thao tác được **sắp xếp** theo một trình tự xác định sao cho sau khi thực hiện dãy thao tác đó, từ **Input** của bài toán, ta nhận được **Output** cần tìm

□ Ví dụ: Thuật toán giải pt $ax + b = 0$

- Nếu $a = 0$
 - $b = 0$ thì phương trình có nghiệm bất kì.
 - $b \neq 0$ thì phương trình vô nghiệm.
- Nếu $a \neq 0$
 - Phương trình có nghiệm duy nhất $x = -b/a$



Al-Khwarizmi (780-850) - người có ảnh hưởng lớn đến sự hình thành thuật ngữ “Algorithm”

Các đặc trưng của thuật toán

- ❑ **Input (dữ liệu vào):** Mỗi thuật toán cần có một số (có thể bằng 0) các dữ liệu ban đầu
- ❑ **Output (Kết quả):** Thuật toán phải cho ra được kết quả
- ❑ **Tính xác định:** Các thao tác phải xác định, không nhập nhằng, lẫn lộn, tùy tiện.
- ❑ **Tính khả thi:** thuật toán phải có khả năng thực hiện được trong một thời gian hữu hạn
- ❑ **Tính kết thúc (tính dừng):** thuật toán phải dừng sau một số hữu hạn bước
- ❑ **Tính phổ dụng:** có thể áp dụng cho một lớp các bài toán có đầu vào tương tự nhau.

2. Các bước xây dựng chương trình



3. Thuật toán và chương trình

- **Chương trình** là tập hợp **dãy các lệnh** điều khiển máy tính thực hiện, hay nói cách khác đó một cách diễn tả **thuật toán** bằng một **ngôn ngữ lập trình** để máy tính có thể hiểu được.
- Các cách biểu diễn thuật toán
 - Sử dụng ngôn ngữ tự nhiên
 - Dùng sơ đồ khối
 - Bằng ngôn ngữ lập trình

Sử dụng ngôn ngữ tự nhiên

Bài toán: Tìm UCLN của hai số nguyên a và b

INPUT: a, b thuộc \mathbb{Z}

OUTPUT: UCLN của a và b

Bước 1. Nhập 2 số nguyên a và b.

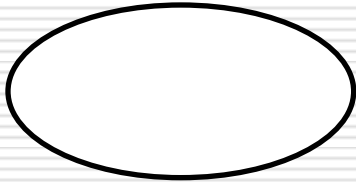
Bước 2. Nếu $a = b$ thì $\text{UCLN} = a$

Bước 3. Nếu $a > b$ thì thay $a = a - b$ quay lại Bước 2

Bước 4. Thay $b = b - a$ quay lại Bước 2

Bước 5. Gán $\text{UCLN} = a$ và kết thúc

Sử dụng sơ đồ khối



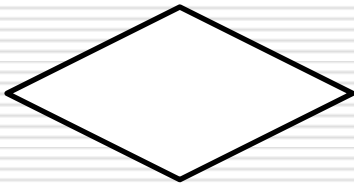
Khối giới hạn

Chỉ thị bắt đầu và kết thúc.



Khối vào ra

Nhập/Xuất dữ liệu.



Khối lựa chọn

Tùy điều kiện sẽ rẽ nhánh.



Khối thao tác

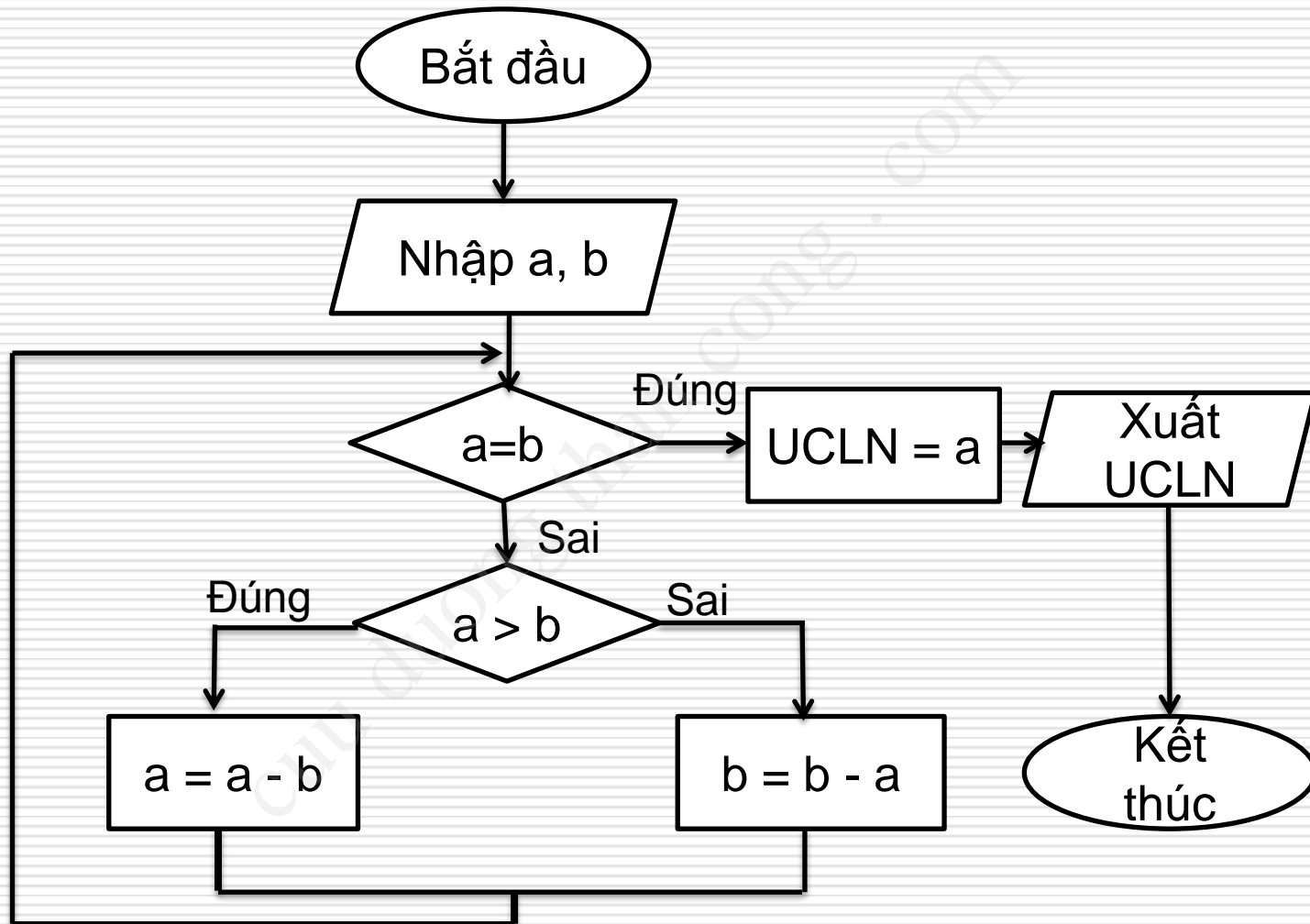
Ghi thao tác cần thực hiện.



Đường đi

Chỉ hướng thao tác tiếp theo.

Sử dụng sơ đồ khối



Cài đặt thuật toán ngôn ngữ lập trình

```
#include <stdio.h>
#include <conio.h>
int a, b;
int main()
{
    printf("Nhap a,b: ");
    scanf("%d%d",&a, &b);
    while (a!=b)
    {
        if (a>b) a=a-b;
        else b=b-a;
    }
    printf("\nUCLN la: %d",a);
    getch();
}
```

Ví dụ về thuật toán

Kiểm tra tính nguyên tố của một số nguyên dương N

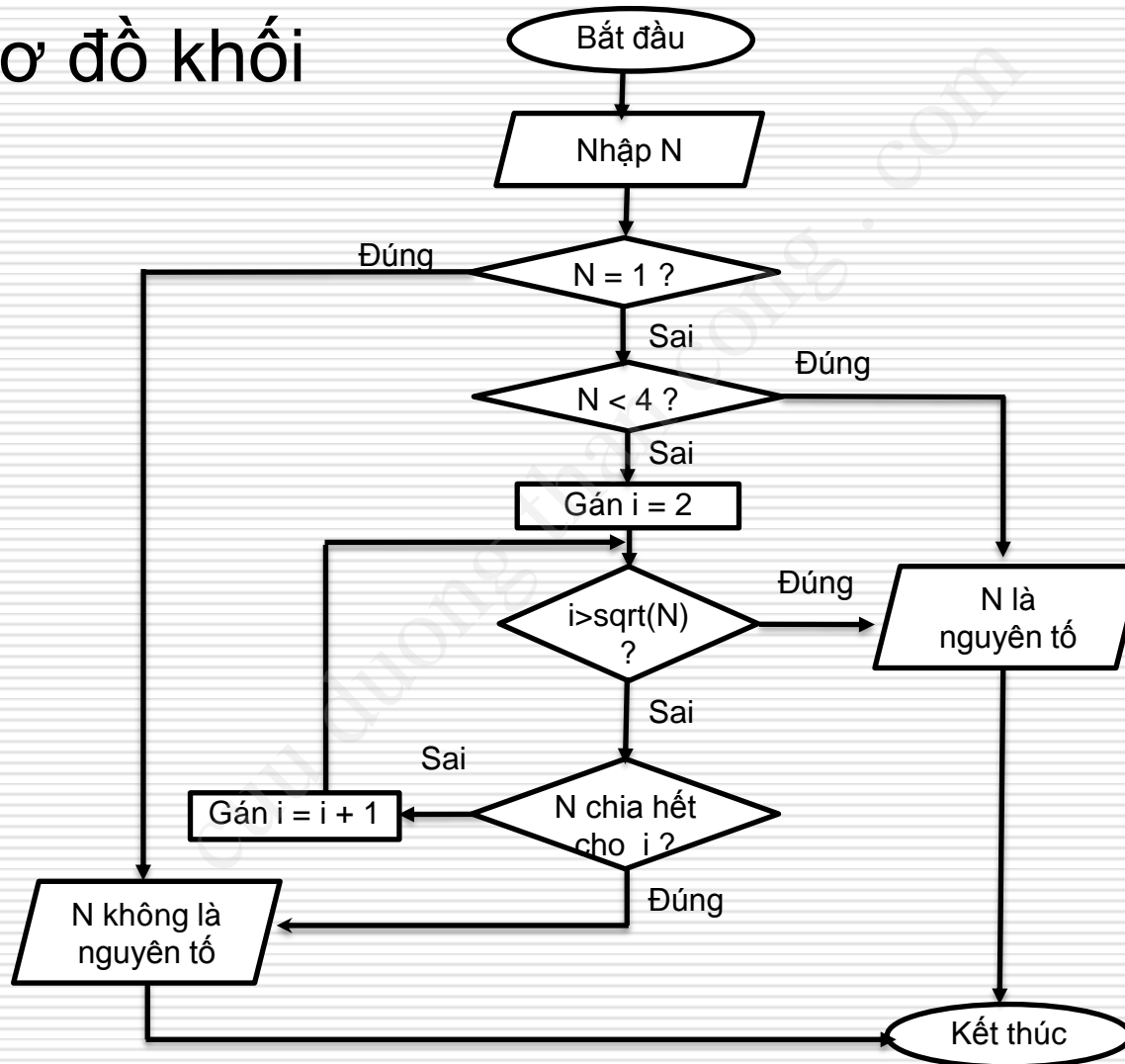
INPUT: N nguyên dương

OUTPUT: N là nguyên tố hay không?

- c 1.** Nhập số nguyên dương N;
- c 2.** Nếu $N = 1$ thì thông báo N không nguyên tố rồi kết thúc;
- c 3.** Nếu $N < 4$ thì thông báo N là nguyên tố rồi kết thúc;
- c 4.** Gán $i = 2$;
- c 5.** Nếu $i > [\sqrt{N}]$ thì thông báo N là nguyên tố rồi kết thúc;
[x] kí hiệu phần nguyên của x, là số nguyên không lớn hơn x và gần x nhất.
- c 6.** Nếu N chia hết cho i thì thông báo N không nguyên tố rồi kết thúc
- c 7.** Gán $i = i + 1$ rồi quay lại bước 5

Ví dụ về thuật toán (tt)

□ Sơ đồ khối



Ví dụ về thuật toán (tt)

Bài toán tìm kiếm

Thuật toán tìm kiếm tuần tự (Sequential Search)

INPUT: Dãy A gồm N số nguyên đôi một khác nhau a_1, a_2, \dots, a_n và số nguyên k
OUTPUT: chỉ số i mà $a_i = k$ hoặc thông báo không có số hạng nào của dãy A có giá trị bằng k

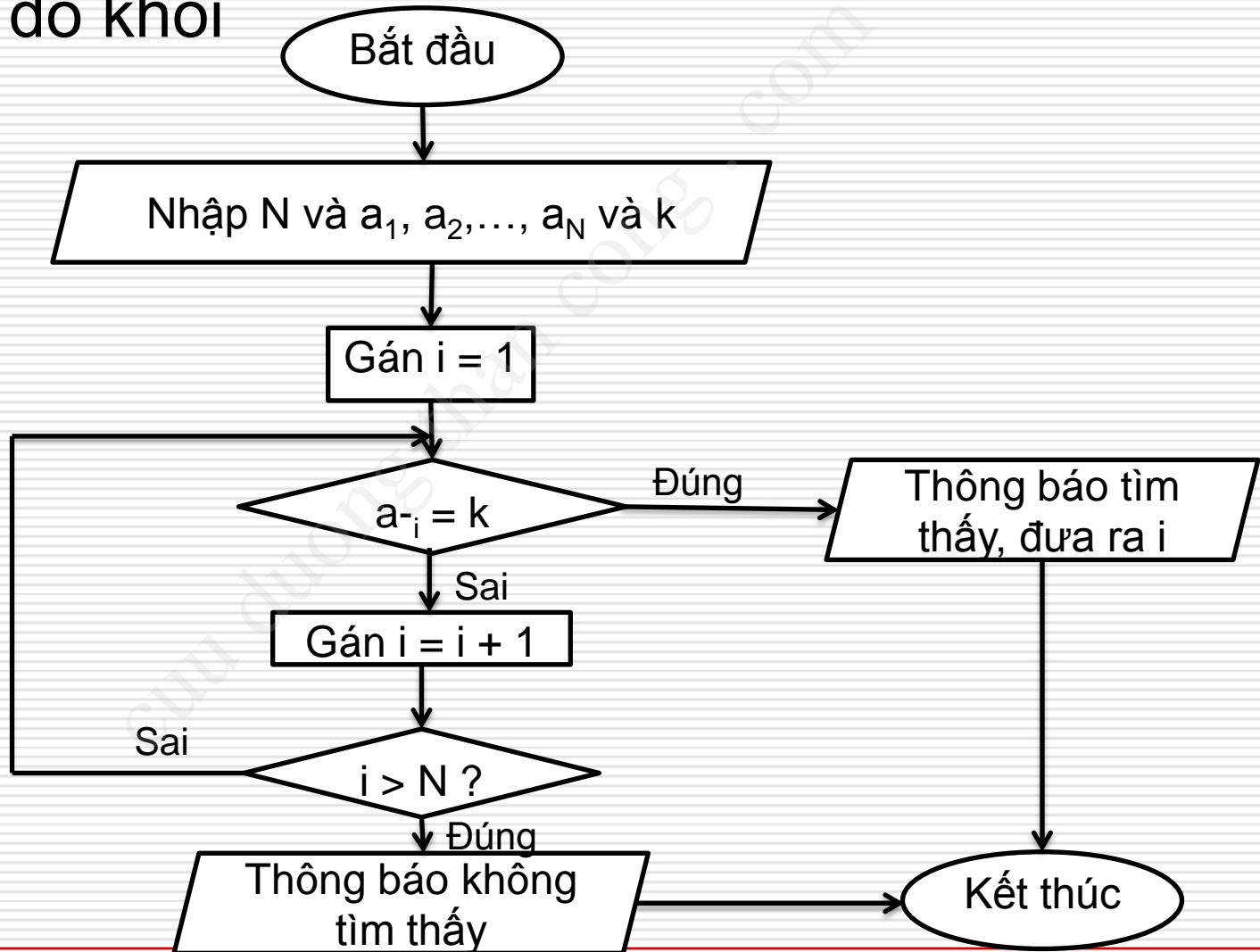
□ Ý tưởng:

■ Lần lượt so sánh các giá trị của dãy với k:

- Nếu có giá trị $a_i = k$ thì đưa ra i
- Nếu khi duyệt hết dãy mà không có giá trị nào bằng k thì đưa thông báo không tìm thấy.

Ví dụ về thuật toán (tt)

❑ Sơ đồ khối



Ví dụ về thuật toán (tt)

Bài toán tìm kiếm

Thuật toán tìm kiếm nhị phân (Binary Search)

INPUT: Dãy A là dãy tăng gồm N số nguyên đôi một khác nhau a_1, a_2, \dots, a_n và số nguyên k

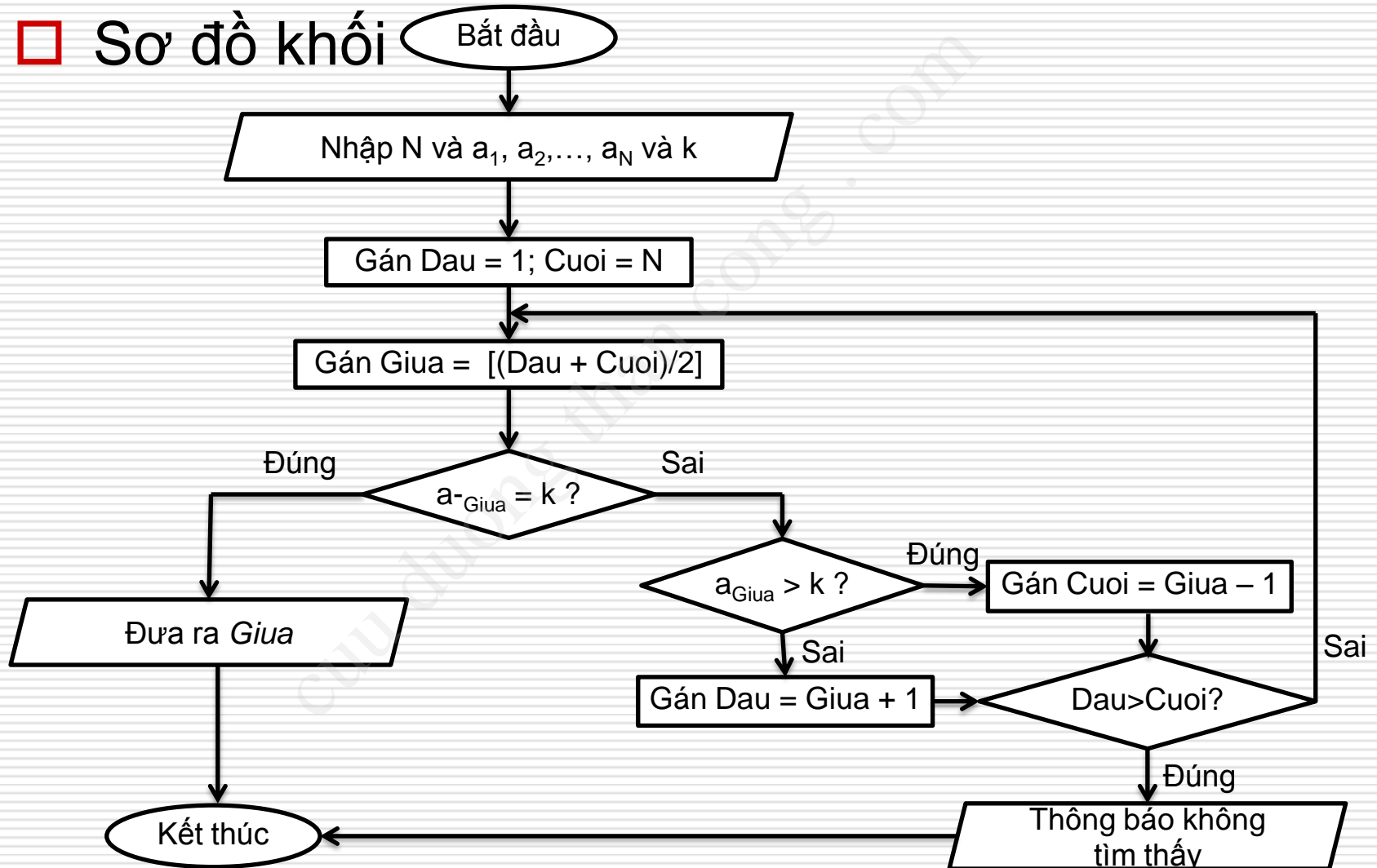
OUTPUT: chỉ số i mà $a_i = k$ hoặc thông báo không có số hạng nào của dãy A có giá trị bằng k

□ Ý tưởng:

- Chọn số hàng a_{Giua} để so sánh với k, trong đó $Giua = [(N+1)/2]$
- Nếu $a_{Giua} = k$ thì Giua là chỉ số cần tìm
- Nếu $a_{Giua} > k$ thì tìm kiếm trên dãy a_1, \dots, a_{Giua-1}
- Nếu $a_{Giua} < k$ thì tìm kiếm trên dãy a_{Giua+1}, \dots, a_N
- Lặp lại cho đến khi tìm thấy k trong dãy hoặc phạm vi tìm kiếm bằng rỗng

Ví dụ về thuật toán (tt)

□ Sơ đồ khối



4. Giới thiệu ngôn ngữ lập trình C

□ Giới thiệu

- Ngôn ngữ C do Dennis Ritchie sáng chế tại Bell Telephone (AT&T) năm 1972 nhằm mục đích viết hệ điều hành Unix
- Tiền thân của ngôn ngữ B, KenThompson, cũng tại Bell Telephone.
- C được viện chuẩn hoá Mỹ (ANSI: American National Standard Institute) làm thành tiêu chuẩn với tên gọi ANSI C năm 1983.
- Là ngôn ngữ lập trình có cấu trúc và phân biệt chữ HOA - thường (case sensitive)

Giới thiệu ngôn ngữ lập trình C (tt)

□ Ưu điểm của C

- **Rất mạnh và mềm dẻo**, có khả năng thể hiện bất cứ ý tưởng nào, dùng viết hệ điều hành, các trình điều khiển, soạn thảo văn bản,..., chương trình dịch
- **Được sử dụng rộng rãi** bởi các nhà lập trình chuyên nghiệp. Chương trình viết bởi C rất hiệu quả (có thể đạt 80% tính năng của chương trình đó viết bằng mã máy)
- **Có tính khả chuyển**, dễ thích nghi, ít thay đổi trên các hệ thống máy tính khác nhau.
- **C có ít từ khoá.**
- **C có cấu trúc modul**, sử dụng chương trình con loại hàm, có thể sử dụng nhiều lần

Giới thiệu ngôn ngữ lập trình C (tt)

❑ Nhược điểm của C

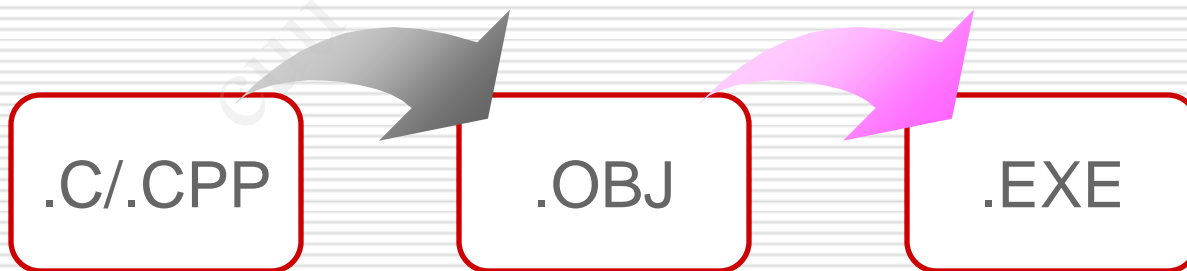
- Cú pháp lạ và khó học
- Một số kí hiệu của C có nhiều nghĩa khác nhau (ví dụ kí hiệu * là toán tử nhân, toán tử không định hướng, thay thế...)
- C quá mềm dẻo (truy nhập tự do vào dữ liệu, trộn lẫn toán tử...)

❑ C là ngôn ngữ bậc trung (medium-level language)

- C kết hợp được các tính năng ngôn ngữ bậc cao với ngôn ngữ bậc thấp
- C mạnh về xử lí bit, địa chỉ ô nhớ → thích hợp lập trình hệ thống

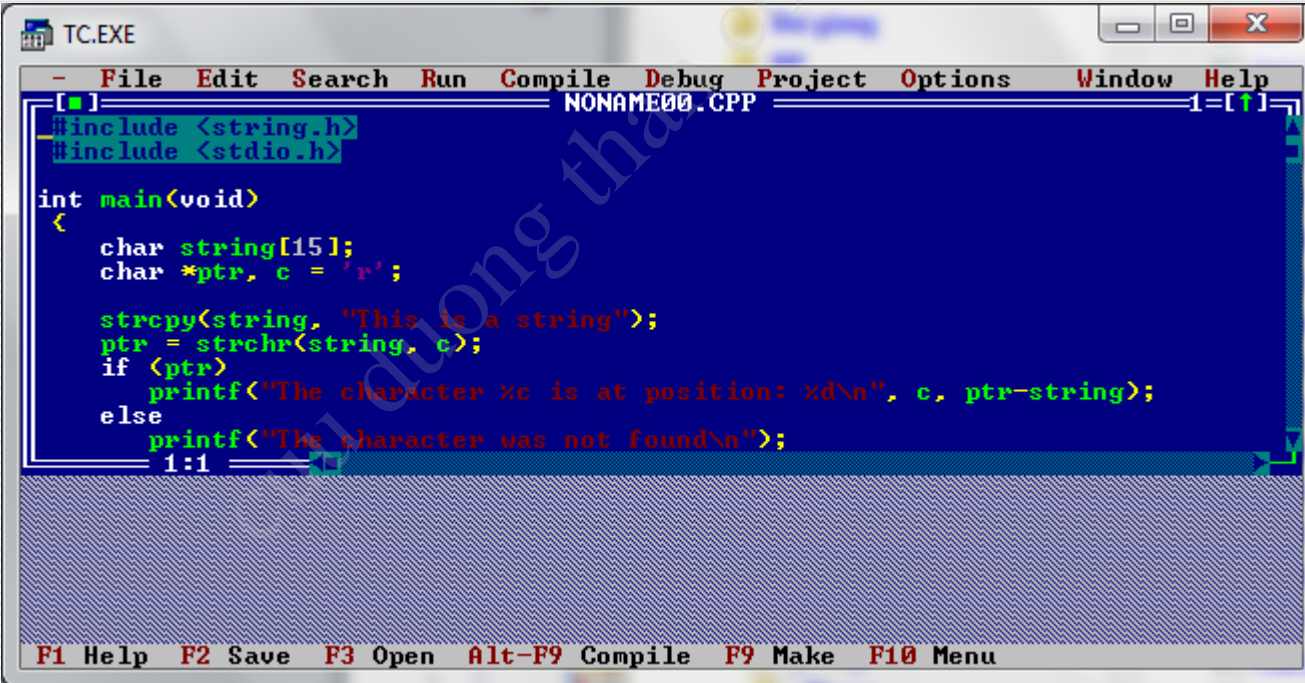
Môi trường lập trình

- ❑ Môi trường phát triển tích hợp IDE (Integrated Development Environment)
 - Biên tập chương trình nguồn (Trình EDIT).
 - Biên dịch chương trình (Trình COMPILE).
 - Chạy chương trình nguồn (Trình RUNTIME).
 - Sửa lỗi chương trình nguồn (Trình DEBUG).



Môi trường lập trình

- ❑ Turbo C++ 3 for DOS.
 - Thực thi file TC\BIN\TC.EXE



The screenshot shows the Turbo C++ 3.0 IDE window titled "TC.EXE". The menu bar includes File, Edit, Search, Run, Compile, Debug, Project, Options, Window, and Help. The status bar at the bottom shows function key shortcuts: F1 Help, F2 Save, F3 Open, Alt-F9 Compile, F9 Make, and F10 Menu. The editor displays a C program named "NONAME00.CPP" with the following code:

```
#include <string.h>
#include <stdio.h>

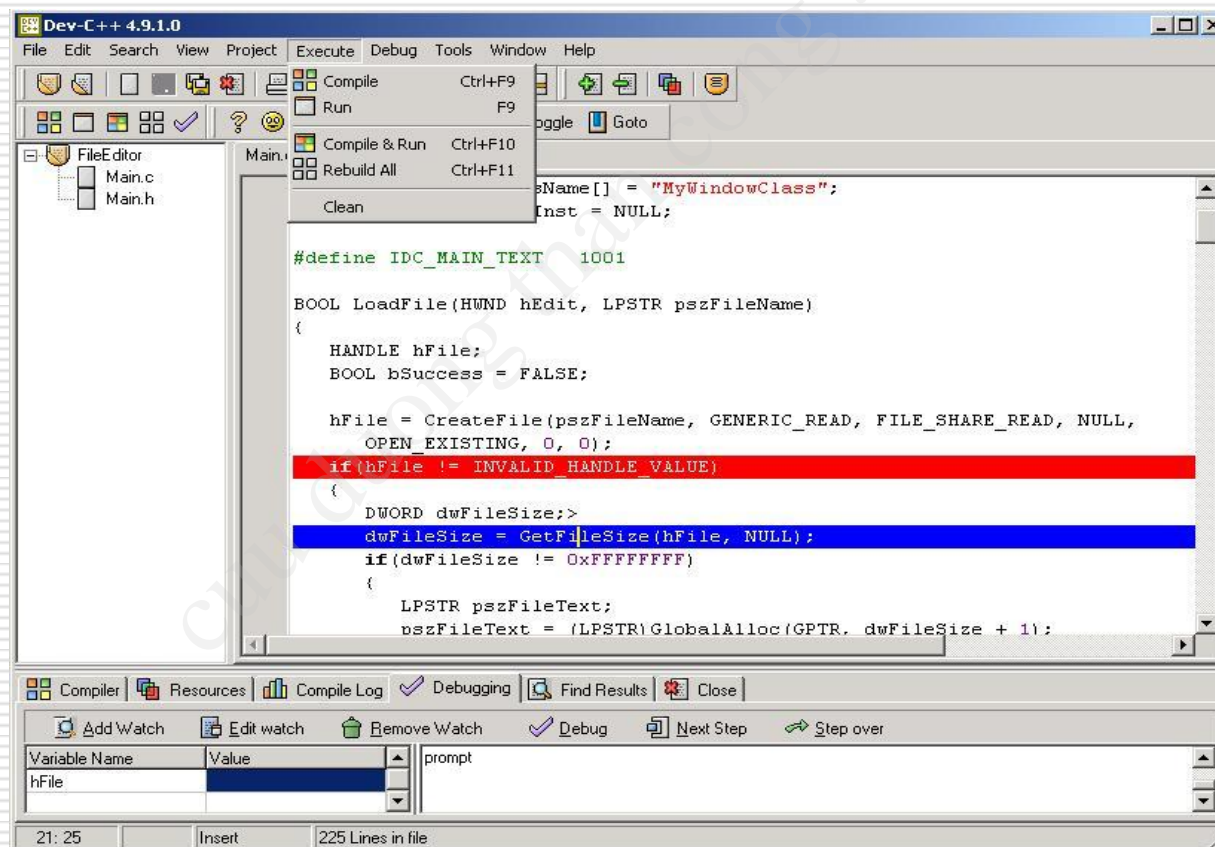
int main(void)
{
    char string[15];
    char *ptr, c = 'r';

    strcpy(string, "This is a string");
    ptr = strchr(string, c);
    if (ptr)
        printf("The character %c is at position: %d\n", c, ptr-string);
    else
        printf("The character was not found\n");
}
```

Môi trường lập trình

❑ Dev-C

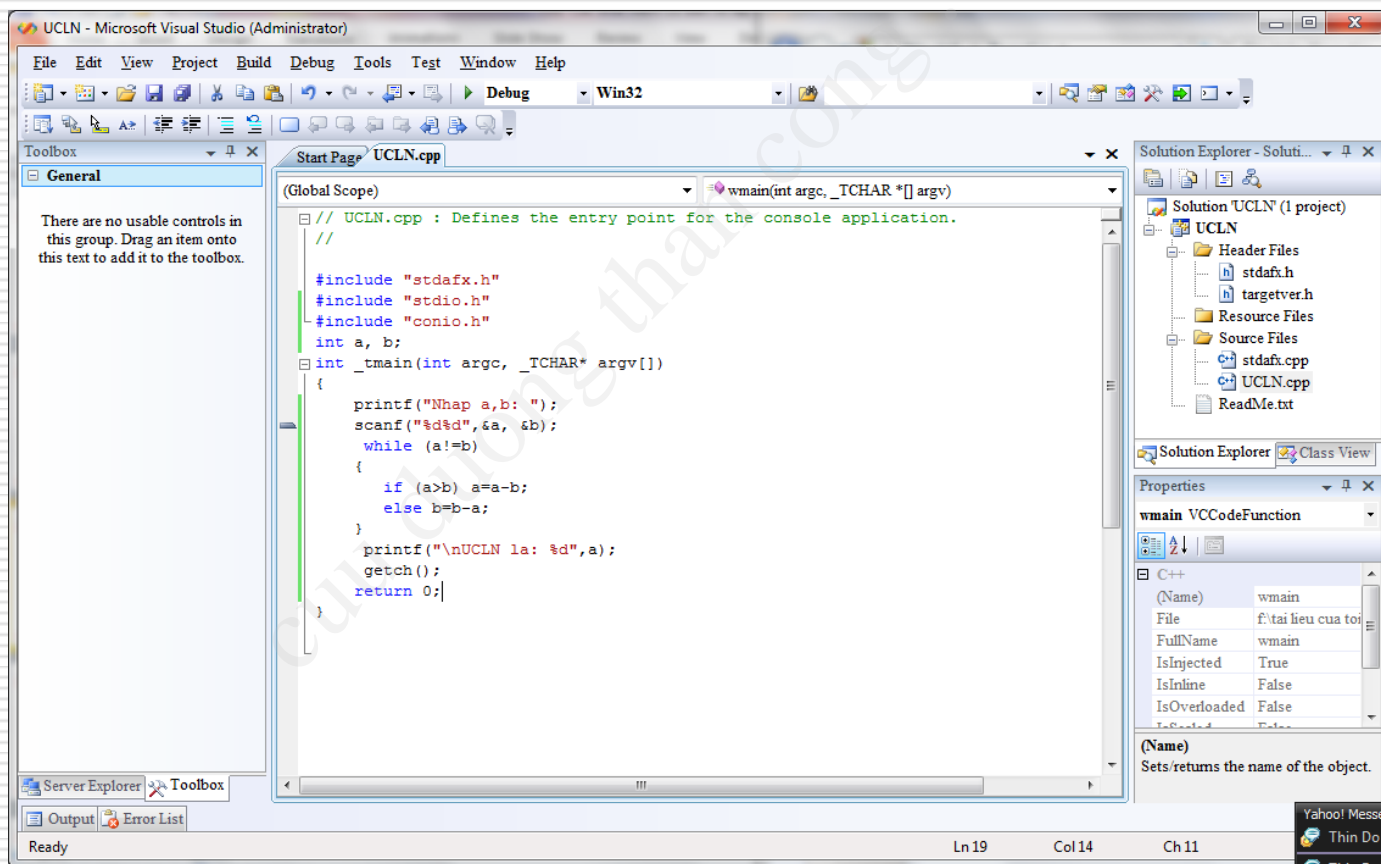
- **Dev-C++ 5.0** (<http://www.bloodshed.net/dev/devcpp.html>)



Môi trường lập trình

❑ Visual Studio

■ VS 6.0, VS2003, VS2005, VS2008, VS2010...



Bài tập lý thuyết

1. Thuật toán là gì? Trình bày các tính chất quan trọng của một thuật toán?
2. Các bước xây dựng chương trình?
3. Các cách biểu diễn thuật toán? Ưu và khuyết điểm của từng phương pháp?
Cho ví dụ minh họa.



Bài tập thực hành



4. Nhập năm sinh của một người. Tính tuổi người đó.



5. Nhập 2 số a và b. Tính tổng, hiệu, tích và thương của hai số đó.



6. Nhập tên sản phẩm, số lượng và đơn giá. Tính tiền và thuế giá trị gia tăng phải trả, biết:

a. $\text{tiền} = \text{số lượng} * \text{đơn giá}$

b. $\text{thuế giá trị gia tăng} = 10\% \text{ tiền}$



Bài tập thực hành



7. Nhập điểm thi và hệ số 3 môn Toán, Lý, Hóa của một sinh viên. Tính điểm trung bình của sinh viên đó.



8. Nhập bán kính của đường tròn. Tính chu vi và diện tích của hình tròn đó.



9. Nhập vào số xe (gồm 4 chữ số) của bạn. Cho biết số xe của bạn được mấy nước?



10. Nhập vào 2 số nguyên.
Tính min và max của hai số đó.

