

# TIN ĐẠI CƯƠNG

---

## BÀI 2: CHƯƠNG TRÌNH C++ ĐẦU TIÊN

# Nội dung buổi trước

---

- Giới thiệu về môn học: giáo trình, nội dung giảng dạy, mục tiêu môn học, cách tính điểm
- Khái niệm chương trình máy tính và cách thức máy tính thực thi một phần mềm
- Các bước để viết một chương trình
- Giới thiệu ngôn ngữ lập trình C++
- Giới thiệu công cụ Dev-C++: bắt đầu chương trình, viết mã, dịch, chạy, sửa lỗi
- Yêu cầu về nhà: cài đặt và làm quen Dev-C++

# Nội dung chính

---

1. Chương trình C++ đơn giản
2. Biến & Định danh
3. Biểu thức, Phép toán & Phép gán
4. Các cấu trúc điều khiển
5. Thuật toán
6. Bài tập

Phần 1

# Chương trình C++ đơn giản

# Chương trình C++ đơn giản

```
#include <iostream>
```

thư viện  
iostream

```
using namespace std;
```

sử dụng tập  
thư viện  
chuẩn

hàm  
chính

```
int main() {
```

```
double x;
```

khai báo  
số thực x

```
cin >> x;
```

nhập x từ  
bàn phím

tính  $x^2$  và  
in ra màn  
hình

```
cout << x * x;
```

```
return 0;
```

trả về 0 cho  
hệ thống

```
}
```

# Chương trình C++ đơn giản

---

Viết chương trình: nhập N, tính và in ra  $\sqrt[2]{N}$

```
#include <iostream>           // sử dụng thư viện vào/ra chuẩn
#include <cmath>               // sử dụng thư viện các hàm toán
using namespace std;         // sử dụng không gian tên chuẩn

int main() {                  // bắt đầu hàm main()
    double n, x;              // khai báo 2 biến n và x
    cout << "N = ";          // in ra màn hình "N = "
    cin >> n;                 // nhập dữ liệu vào biến n
    x = sqrt(n);              // tính giá trị của x
    cout << "X = " << x;     // in ra màn hình: "X = ..."
}
```

Phần 2

# Biến & Định danh

# Biến và định danh

---

- Quy tắc khai báo chung:

<kiểu> <tên biến>;

<kiểu> <tên biến> = <giá trị>;

- Ví dụ:

```
int x;           // số nguyên x
int n = 100;     // số nguyên n, giá trị 100
double d = 1.5; // số thực d, giá trị 1,5
double m;       // số thực m
bool kiểmtra;   // biến logic kiểmtra
bool ok = false; // biến logic ok, giá trị = sai
int a, b, c;    // 3 số nguyên a, b, c
```



# Biến và định danh

---

- **Khái niệm:** vùng trong bộ nhớ máy tính dùng để chứa những kết quả tính toán
  - Cần được đặt tên để dễ thao tác
  - Gọi là "biến" (variable)
- **Nguyên tắc:**
  - Phải khai báo trước khi dùng
  - Phải chỉ ra kiểu (loại số)
  - Viết liền, phân biệt chữ hoa và chữ thường
  - Bắt đầu bởi chữ cái (tiếng Anh) theo sau là chữ cái hoặc chữ số: **n**, **soA**, **thamso1**, **diepvien007**,...

# Biến và định danh

---

- Không chỉ cho phép đặt tên biến, C++ còn cho phép đặt tên cho các thành phần khác:
  - Tên hàm (function)
  - Tên lớp (class)
  - Tên cấu trúc (struct)
- Định danh (identifier) là quy cách chung cho việc đặt tên của các thành phần trên
  - Có thể sử dụng thêm dấu gạch dưới (  )
  - Không được trùng với từ khóa của ngôn ngữ
  - Không được trùng nhau

Phần 3

# Biểu thức, Phép toán & Phép gán

# Biểu thức (expression)

---

- **Khái niệm:** sự kết hợp giữa các giá trị, biến, phép toán và các cặp ngoặc để có thể thực hiện tính toán được kết quả cụ thể nào đó
  - Nhận xét: tương tự như biểu thức trong toán học

- **Ví dụ:**

$$10 - 20 * 3$$

$$n + 2 * m$$

$$m * -1 / ( k + 1.5 )$$

$$(dayNho + dayLon) * chieuCao / 2$$

$$(a + b + c) / 3$$

# Phép toán (operator)

---

- **Phép toán:** C++ cho phép sử dụng hầu hết các phép toán thông dụng
  - Phép tính số học: cộng (+), trừ (-), nhân (\*), chia lấy thương (/), chia lấy dư (%)
  - Phép so sánh, phép toán logic,...
- Thứ tự thực hiện có sự ưu tiên tương tự như trong toán học
- Các hàm toán học, dùng trong những trường hợp phức tạp hơn: căn 2 (sqrt), mũ (power),...  
*(sẽ giới thiệu đầy đủ hơn ở phần bàn về kiểu dữ liệu)*

# Phép gán

---

- **Định nghĩa:** phép toán ghi nhớ giá trị biểu thức vào một biến
- **Cú pháp:**  
`<biến> = <biểu thức>;`
- **Ví dụ:**  
`n = 10;`  
`m = n + 5 / 3;`  
`t = a + b + c;`
- **Chú ý:** khi viết mã, rất dễ nhầm phép gán (=) và phép so sánh bằng (==)

Phần 4

# Các cấu trúc điều khiển

# Các cấu trúc điều khiển

---

- Có 3 cấu trúc điều khiển cơ bản
  - **Tuần tự**: Thực hiện tuyến tính từng việc một
  - **Lặp**: Thực hiện lặp lại một hoặc nhiều việc cho đến khi điều kiện nhất định được thỏa mãn
  - **Lựa chọn (rẽ nhánh)**: Chọn thực hiện một hoặc nhiều việc dựa trên một điều kiện nhất định
- Các cấu trúc này tương tự như nhiều hành vi trong cuộc sống
- Đọc trước các chương 2, 3 và 4 của giáo trình để tìm hiểu về các điều khiển này



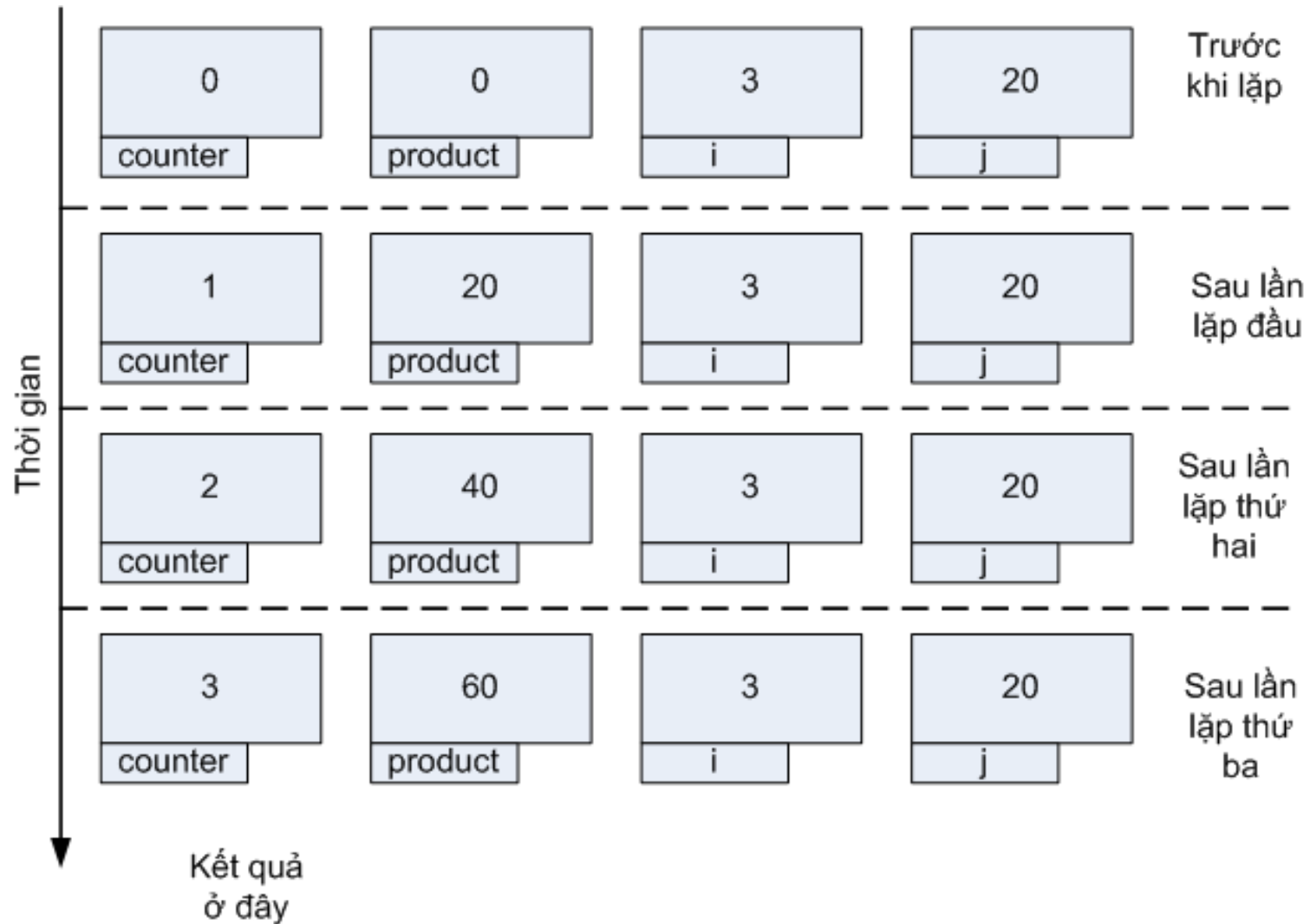
# Ví dụ minh họa

---

(*phần 1.1.2 của giáo trình*) Nhân 2 số tự nhiên  $i$  và  $j$  trên máy tính không có phép nhân

1. Đặt biến **product = 0**
2. Đặt biến **counter = 0**
3. Lặp lại bước 4 và 5 chừng nào **counter < i**:
4.       Đặt product bằng chính nó cộng với  $j$
5.       Tăng biến counter lên 1
6. Trả về kết quả là giá trị của product

# Ví dụ minh họa



# Ví dụ mở rộng

---

Nhân 2 số nguyên  $i$  và  $j$  trên máy tính không có phép nhân (bỏ điều kiện  $i$  và  $j$  không âm)

**if** cả  $i$  và  $j$  đều không âm **then**

- sử dụng thuật toán nhân không âm  $i$  và  $j$  và lưu kết quả trong `product`

**else if** cả  $i$  và  $j$  đều âm **then**

- đổi dấu cả  $i$  và  $j$ , lúc này chúng đều không âm
- sử dụng thuật toán nhân không âm  $i$  và  $j$  và lưu kết quả trong `product`

**else**

# Ví dụ mở rộng

---

- **if  $i < 0$  then**
  - đổi dấu của  $i$
- **else**
  - đổi dấu của  $j$
- **end if**
  - sử dụng thuật toán nhân không âm  $i$  và  $j$  và lưu kết quả trong `product`
  - đảo dấu của `product`

**end if**

**return product**

Phần 5

# Thuật toán

# Định nghĩa và đặc trưng

---

- **Định nghĩa**: các bước cần tiến hành để giải quyết một công việc cụ thể nào đó
- Thuật toán phổ biến trong cuộc sống, có trước máy tính và có nhiều dạng khác nhau
- Đặc trưng (*xem phần 1.1 của giáo trình*):
  - Tính hữu hạn
  - Tính máy móc
  - Tính dừng
  - Mở rộng: Tính đúng
  - Mở rộng: Tính tổng quát

# Ví dụ đơn giản

---

Ví dụ: Tính bình phương của số  $m$

Bước 1: Nhập giá trị cho  $m$

Bước 2: Tính giá trị  $m \times m$  và đưa vào  $s$

Bước 3: Trả về giá trị  $s$  cho chương trình gọi

Ta thấy:

- Thuật toán có 3 bước
- Viết rõ ràng, không thể hiểu sai
- Làm theo thuật toán ta có thể tính được kết quả mà không cần hiểu khái niệm “bình phương” là gì

# Ví dụ phức tạp hơn

---

Giải phương trình  $ax^2 + bx + c = 0$  (với  $a \neq 0$ )

- Bước 1: Nhập các giá trị  $a, b, c$
- Bước 2: Nếu  $a = 0$  thì thông báo lỗi và dừng
- Bước 3: Tính  $d = b^2 - 4 \times a \times c$
- Bước 4: Nếu  $d < 0$  thì thông báo vô nghiệm và dừng
- Bước 5: Nếu  $d > 0$  thì thực hiện Bước 7
- Bước 6: Thông báo có nghiệm  $x = -b/2/a$  và dừng
- Bước 7: Thông báo có hai nghiệm
  - $x_1 = (-b + \sqrt{d})/2/a$
  - $x_2 = (-b - \sqrt{d})/2/a$



# Ví dụ vui

---

Cho con sư tử vào tủ lạnh

Bước 1: Mở cửa tủ lạnh

Bước 2: Cho con sư tử vào

Bước 3: Đóng cửa tủ lạnh

**Câu hỏi:** *hãy nêu một vài thuật toán tương tự trong cuộc sống (thuật toán nhưng không dính dáng gì tới máy tính)*

Phần 6

# Bài tập

# Bài tập

---

1. Hãy xây dựng thuật toán (hoặc viết chương trình) giải phương trình bậc nhất  $P(x): a x + b = c$
2. Hãy xây dựng thuật toán để tính tổng các chữ số của một số nguyên  $N$  bất kỳ.  
Ví dụ:  $N = 2016$  thì thuật toán trả về 9 ( $2+0+1+6$ )
3. Làm thế nào để tạo các số ngẫu nhiên trong máy tính? Hãy xây dựng một thuật toán để có thể tạo ra được các số như vậy.
4. Nhập điểm toán, lý và hóa của một học sinh. In ra điểm trung bình 3 môn học của học sinh đó.