

Chương 2

Các kiểu dữ liệu và khai báo

*Khoa công nghệ thông tin
Trường Đại học mở tphcm*

1

Mục tiêu

- Cách sử dụng định danh khi đặt tên cho biến, hằng, hàm trong C++
- Các kiểu dữ liệu cơ bản trong C++ và phạm vi biểu diễn.
- Cách định nghĩa, khởi tạo biến sử dụng biến và hằng trong C++.

2

Nội dung

1. Danh hiệu và từ khóa
2. Các kiểu dữ liệu cơ bản
3. Biến
4. Hằng
5. Một số hằng định nghĩa trước trong thư viện C++

3

2.1 Danh hiệu và từ khóa

- **Từ khóa** hay **từ dành riêng** (keywords/reserved words) là các từ có ý nghĩa đặc biệt đối với chương trình dịch.
- Ví dụ một số từ khóa trong C++: `int double char const void return ...`
- Các ký tự trong từ khóa C++ là chữ thường.
- Không được dùng từ khóa để đặt tên cho biến, hàm, ... trong chương trình.

4

Danh hiệu và từ khóa (tt)

- **Tên** (name) hay danh hiệu, định danh (identifier) là tên biến, tên hằng, tên hàm, tên kiểu dữ liệu do người lập trình định nghĩa và được đặt theo luật.
- Luật đặt tên trong C++:
 - Tên thường bắt đầu bằng ký tự, \$, _
 - Tên chỉ gồm ký tự chữ, số, dấu gạch dưới (_), dấu \$. Không được phép có khoảng trắng và các ký tự khác (dấu chấm, dấu phẩy, %...)
 - Tên không được trùng với từ khóa.
 - Tên trong C++ phân biệt chữ thường và hoa (case sensitive).

Danh hiệu và từ khóa (tt)

- Ví dụ các tên hợp lệ:
 - x
 - Gross\$Pay
 - number_of_studentes
 - n2
 - Polygon
 - PI
 - TAX_RATE
- Ví dụ các tên không hợp lệ:
 - 2x starttime
 - Start menu
 - RATE%
 - 2018Sales
- C++ phân biệt ký tự hoa và thường, x và X là hai tên khác nhau.

Danh hiệu và từ khóa (tt)

- Lưu ý khi đặt tên:
 - Nên chọn tên có ý nghĩa, giúp người đọc có thể hiểu chương trình một cách dễ dàng.
 - Không nên viết tắt vì sẽ gây khó hiểu:
mtbf TLA myw nbv
 - Các tên ngắn gọn đã được quy ước:
x: biến cục bộ
i: chỉ số của vòng lặp
 - Không nên dùng tên quá dài:
the_number_of_elements
remaining_free_slots_in_the_symbol_table
 - Nên đặt tên ngắn gọn, có nghĩa:
partial_sumelement_count staple_partition

Các kiểu dữ liệu và khai báo

7

7

Quy ước đặt tên

- Tên biến: lowerCamelCase
double centimeter;
int totalInches;
- Tên hàm: lowerCamelCase, bắt đầu bằng động từ.
double calculateAverage();
void run();
void print();
- Tên kiểu dữ liệu do người dùng định nghĩa: UpperCamelCase
Graph
Square
- Tên hằng: ký tự in hoa, dùng dấu gạch dưới phân cách các từ.
const int MAX_PARTICIPANTS = 10;



Các kiểu dữ liệu và khai báo

8

8

2.2 Các kiểu dữ liệu cơ bản

- Dữ liệu có thể là một ký tự, một số hoặc một đoạn văn bản...
- Khi viết chương trình ta phải cho máy tính biết kiểu dữ liệu (data type) cần xử lý.
- Các ngôn ngữ lập trình đều sử dụng hai kiểu dữ liệu cơ bản:
 - **Dữ liệu số** (numeric data type): số nguyên (integer) và số dấu chấm động (floating point).
 - **Dữ liệu chuỗi ký tự** (character string data type): ký tự (character) và chuỗi (string).
- Ngoài ra, các ngôn ngữ lập trình còn có kiểu dữ liệu luận lý (boolean data type).

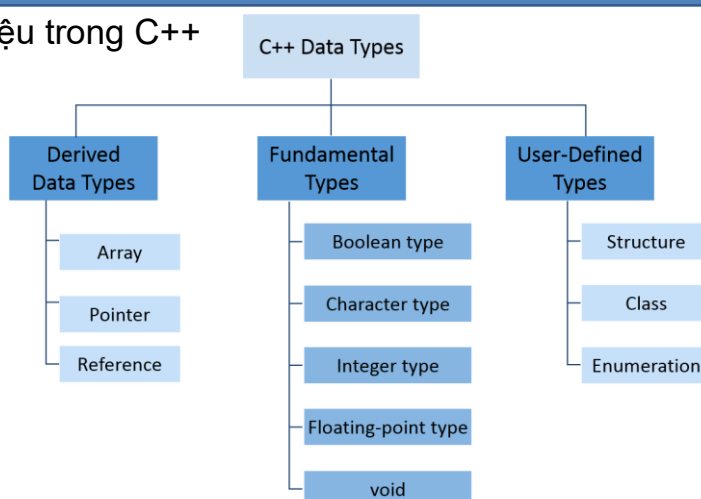
Các kiểu dữ liệu và khai báo

9

9

Các kiểu dữ liệu cơ bản (tt)

- Kiểu dữ liệu trong C++



Các kiểu dữ liệu và khai báo

10

10

Các kiểu dữ liệu cơ bản (tt)

Kiểu (type)	Mô tả	Phạm vi (range)
char	số nguyên 1-byte	0...255
unsigned char	số nguyên không dấu 1-byte	0...255
signed char	số nguyên có dấu 1-byte	-128...127
short	số nguyên 2-byte	-32,768...32,767
unsigned short	số nguyên không dấu 2-byte	0...65,535
int	số nguyên 4-byte	xấp xỉ ± 2 tỷ
unsigned int	số nguyên không dấu 4-byte	xấp xỉ 4 tỷ
long	số nguyên 4-byte	xấp xỉ ± 2 tỷ
unsigned long	số nguyên không dấu 4-byte	xấp xỉ 4 tỷ
bool	lưu giá trị luận lý	true hoặc false
float	số thực chính xác 7 chữ số	3.4×10^{-38} ... 3.4×10^{38}
double	số thực chính xác 15 chữ số	1.8×10^{-308} ... 1.8×10^{308}

Các kiểu dữ liệu và khai báo

11

11

Các kiểu dữ liệu cơ bản (tt)

• bool

- Kích thước: 1 byte.
- Lưu giá trị true (khác 0) hoặc false (0).

• char

- Kích thước: 1 byte.
- Lưu trữ các ký tự chữ, ký tự số và ký tự đặc biệt trong bảng mã ASCII (0-255).
- Ký tự đặt giữa hai dấu nháy đơn:
 - 'A', 'a', '0', ...
 - '*', '+', '&', ...
 - ' ' (character literals)

Một số ký tự đặc biệt:

'\n': newline
 '\\': backslash
 '\t': tab
 '\': single quote
 '\b': backspace
 '\"': double quote
 '\': escape character

Các kiểu dữ liệu và khai báo

12

12

Các kiểu dữ liệu cơ bản (tt)

- **int**

- Kích thước: 4 byte.
- Lưu các giá trị từ -2,147,483,648 đến 2,147,483,647.
- Giá trị cực đại: INT_MAX
- Giá trị cực tiểu: INT_MIN
- Lưu ý khi ghi các giá trị số nguyên (number literals):
 - Số nguyên dương không cần dấu +.
 - Không có dấu phẩy (,) trong số nguyên.
- Ví dụ: -6728 -67 0 78 35267 763

- **double**

- Kích thước: 8 byte
- Lưu các giá trị từ 1.7×10^{-308} đến 1.7×10^{308} (15 chữ số).
- C++ biểu diễn số thực dùng dấu chấm động.
- Ví dụ: 1.23 .23 0.23 1.0 1.2e10 1.23e-15

Các kiểu dữ liệu cơ bản (tt)

- **Kiểu string (class string):** lưu trữ chuỗi ký tự

- Đây là kiểu dữ liệu có sẵn trong thư viện.
- Chuỗi ký tự có từ 0 hoặc nhiều ký tự và được đặt giữa hai dấu nháy kép.
 - "It is a beautiful day."
 - "Mickey"
 - ""
- Độ dài chuỗi tùy ý.
- Để sử dụng kiểu string ta phải thêm chỉ thị:
 - #include <string>

Các kiểu dữ liệu cơ bản (tt)

- Một số toán tử tương ứng với kiểu dữ liệu

	bool	char	int	double	string
gán	=	=	=	=	=
cộng			+	+	
nối chuỗi					+
trừ			-	-	
nhân			*	*	
chia			/	/	
chia lấy dư			%		
tăng 1			++	++	
giảm 1			--	--	
tăng n			+=n	+=n	
thêm vào cuối					+=
giảm n			-=n	-=n	

Các kiểu dữ liệu và khai báo

15

15

Các kiểu dữ liệu cơ bản (tt)

- Một số toán tử tương ứng với kiểu dữ liệu (tt)

	bool	char	int	double	string
nhân và gán			*=	*=	
chia và gán			/=	/=	
chia dư và gán			%=		
đọc từ s vào x	s >> x	s >> x	s >> x	s >> x	s >> x
ghi từ x ra s	s << x	s << x	s << x	s << x	s << x
so sánh bằng	==	==	==	==	==
không bằng	!=	!=	!=	!=	!=
lớn hơn	>	>	>	>	>
lớn hơn hoặc bằng	>=	>=	>=	>=	>=
nhỏ hơn	<	<	<	<	<
nhỏ hơn hoặc bằng	<=	<=	<=	<=	<=

Các kiểu dữ liệu và khai báo

16

16

2.3 Biến

- Biến (variable) là một vùng nhớ dùng để lưu trữ một giá trị dữ liệu. Mỗi biến có tên (name) và kiểu dữ liệu (type).
- Dữ liệu được lưu trữ trong biến gọi là giá trị (value) của biến.

tuoi ten

- Khi thực hiện, chương trình có thể đọc, ghi, hoặc thay thế các giá trị của biến.
- C++ yêu cầu phải khai báo biến trước khi sử dụng.
- Để khai báo một biến, ta cần phải biết kiểu dữ liệu (data type) mà chương trình sẽ sử dụng (số, chuỗi...).

Các kiểu dữ liệu và khai báo

17

17

Biến (tt)

- Cú pháp khai báo biến:
`dataType variableName;`
 - dataType: kiểu dữ liệu (có trong C++, là kiểu cơ bản hoặc kiểu do người dùng tạo).
 - variableName: tên biến (theo luật đặt tên của C++).

- Ví dụ:

int soluong; soluong

double tyle; tyle

char loiSP; loiSP

string tenSP; tenSP

- Có thể khai báo nhiều biến có cùng kiểu dữ liệu:
`int a, b, c, d;`

Các kiểu dữ liệu và khai báo

18

18

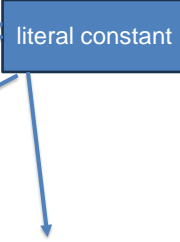
Biến (tt)

- Cú pháp khai báo biến và khởi tạo giá trị:

`dataType variableName = initialValue;`

- Ví dụ:

`int soluong = 5;` soluong 5
`double tyle = 0.05;` tyle 0.05
`char loaiSP = 'K';` loaiSP 'K'
`string tenSP = "Kem danh rang";` tenSP "Kem danh rang"



Biến (tt)

- Ví dụ chương trình in giá trị các biến:

```
#include <iostream>
using namespace std;
int main()
{
    int soTao = 5;
    int soCam = 20;
    int tongcong = soTao + soCam;
    cout << "So trai tao: " << soTao << endl;
    cout << "So trai cam: " << soCam << endl;
    cout << "Tong cong: " << tongcong << endl;
    return 0;
}
```

Biến (tt)

- Một số lỗi thường gặp
 - Sử dụng biến chưa khai báo.
`double x = 12 * y; //lỗi vì biến y chưa khai báo`
`double y = 0.0296;`
 - Sử dụng biến chưa được khởi tạo giá trị.
`double x;`
`double y = x * 2;`
 - Gán giá trị khác kiểu cho biến.
`string hoten = 39; //lỗi`
`int n = "Hoa"; //lỗi`

Các kiểu dữ liệu và khai báo

21

21

2.4 Hằng

- Hằng (named constant) là một vùng nhớ dùng để lưu trữ một giá trị dữ liệu không thay đổi trong suốt thời gian chương trình thực thi.
- Cú pháp định nghĩa hằng:


```
const dataType constantName = value;
```

 - dataType: kiểu dữ liệu
 - constantName: tên hằng (theo luật đặt tên của C++), thường dùng chữ in hoa.
 - value: giá trị phù hợp với kiểu dữ liệu.
- Ví dụ:


```
const double PI = 3.141593;
const double MAX_SIZE = 2.54;
const char BLANK = ' ';
```

Các kiểu dữ liệu và khai báo

22

22

Hàng (tt)

- Một số hằng định nghĩa sẵn trong thư viện:

```
#include <limits.h>
CHAR_BIT      8
CHAR_MIN      -128 (0 if /J option used)
CHAR_MAX      127 (255 if /J option used)

INT_MIN        (-2147483647 - 1)
INT_MAX        2147483647
UINT_MAX       0xffffffff
LONG_MIN       (-2147483647L - 1)
LONG_MAX       2147483647L

#include <float.h>
DBL_MIN        2.2250738585072014e-308
DBL_MAX        1.7976931348623158e+308
DBL_EPSILON    2.204460492503131e-016
```

Các kiểu dữ liệu và khai báo

23

23

Bài tập

- Hãy cho biết các tên nào là hợp lệ trong C++:

1. number
2. Double
3. include
4. AR??EA
5. radius
6. string2
7. 2xyx
8. first-name
9. Last_name
10. _new

Các kiểu dữ liệu và khai báo

24

24

Bài tập

- Cho chương trình tính diện tích và chu vi hình tròn, biết bán kính là giá trị phải nhập.
 - Hãy cho biết cần có bao nhiêu biến, bao nhiêu hằng.
 - Vẽ lưu đồ mô tả thuật giải của chương trình.

Q & A