

ĐẠI NAM
UNIVERSITY

Phần trình bày của:

ĐẬU HẢI PHONG

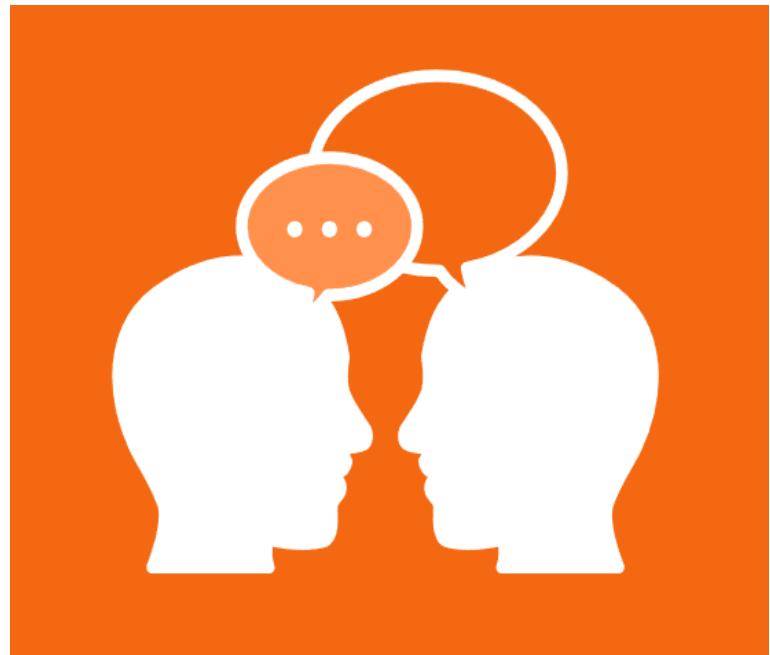
Giảng viên

Đại Nam, ngày 01 tháng 19 năm 2023



LƯU Ý

KHÔNG NÓI
CHUYỆN RIÊNG



KHÔNG SỬ DỤNG
ĐIỆN THOẠI



KHÔNG NGỦ GẬT



GHI CHÉP ĐẦY ĐỦ





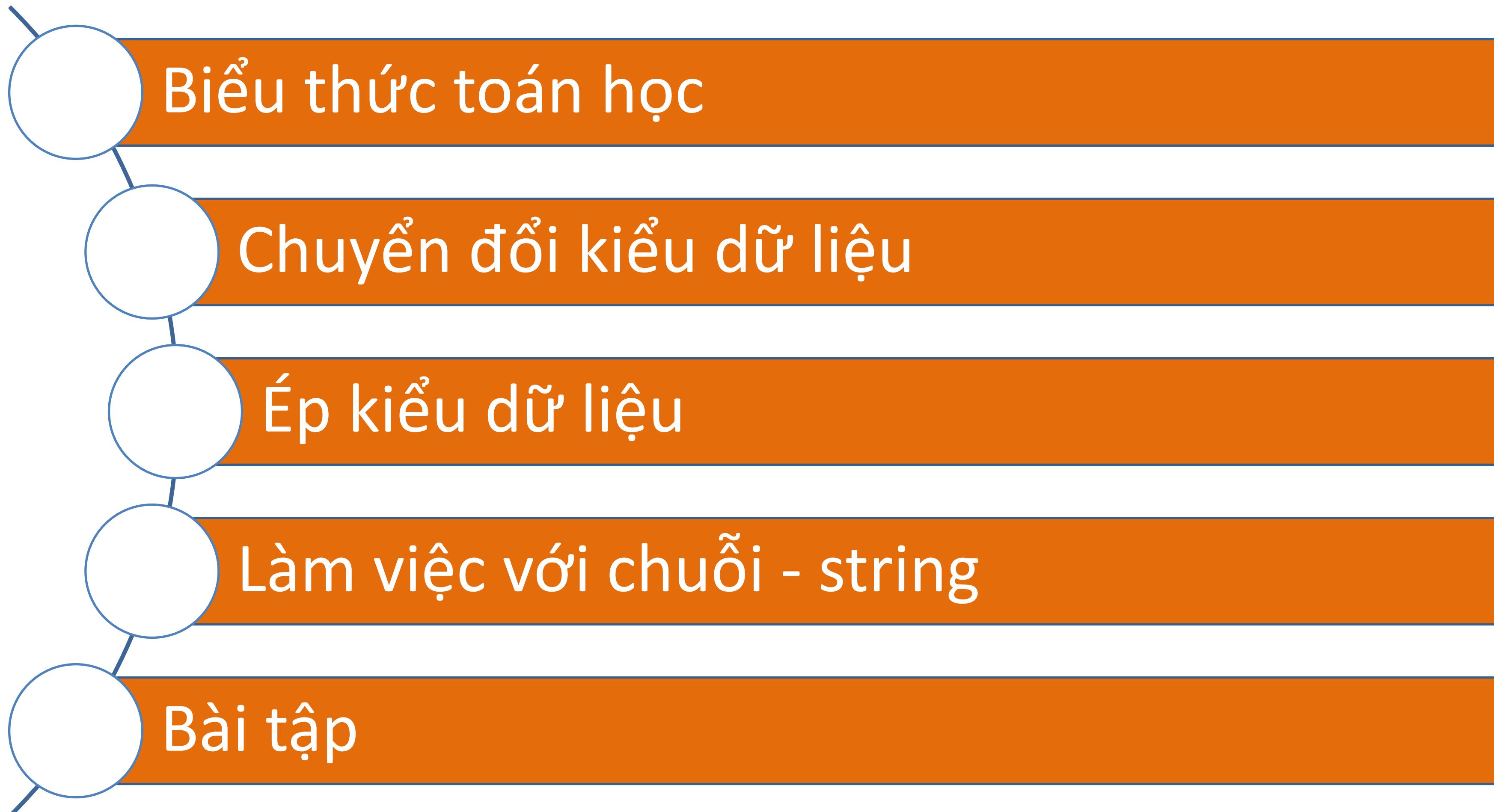
LẬP TRÌNH CƠ BẢN

CHƯƠNG 2

BIỂU THỨC VÀ CÁC THAO TÁC CƠ BẢN

NỘI DUNG

HỌC ĐỂ THAY ĐỔI



cin & vấn đề liên quan (1)

- cin:
 - là đối tượng nhập giá trị đầu vào
 - Yêu cầu có **#include<iostream>**
 - Dùng để nhập giá trị đầu vào từ bàn phím
 - Thông tin truy xuất từ cin với **>>**
 - Đầu vào được lưu trữ trong 1 hoặc nhiều biến
 - VD:
 - int height, length;
 - cout << "How tall is the room? ";
 - cin >> height;
 - cin chuyển đổi dữ liệu phù hợp với biến
 - Nên có lời nhắc đe người dùng biết nhập dữ liệu
 - Có thể nhập nhiều giá trị cùng lúc
 - VD: cin >> height >> width;

cin & vấn đề liên quan (2)

Program 3-2

```
1 // This program asks the user to enter the length and width of
2 // a rectangle. It calculates the rectangle's area and displays
3 // the value on the screen.
4 #include <iostream>
5 using namespace std;
6
7 int main()
8 {
9     int length, width, area;
10
11    cout << "This program calculates the area of a ";
12    cout << "rectangle.\n";
13    cout << "Enter the length and width of the rectangle ";
14    cout << "separated by a space.\n";
15    cin >> length >> width;
16    area = length * width;
17    cout << "The area of the rectangle is " << area << endl;
18    return 0;
19 }
```

Program Output with Example Input Shown in Bold

This program calculates the area of a rectangle.

Enter the length and width of the rectangle separated by a space.

10 20 [Enter]

The area of the rectangle is 200

Program 3-3

```
1 // This program demonstrates how cin can read multiple values
2 // of different data types.
3 #include <iostream>
4 using namespace std;
5
6 int main()
7 {
8     int whole;
9     double fractional;
10    char letter;
11
12    cout << "Enter an integer, a double, and a character: ";
13    cin >> whole >> fractional >> letter;
14    cout << "Whole: " << whole << endl;
15    cout << "Fractional: " << fractional << endl;
16    cout << "Letter: " << letter << endl;
17    return 0;
18 }
```

Program Output with Example Input Shown in Bold

Enter an integer, a double, and a character: **4 5.7 b [Enter]**

Whole: 4

Fractional: 5.7

Letter: b

Biểu thức toán học (1)

- Có thể tạo biểu thức toán học bằng nhiều toán tử toán học
- Biểu thức có thể là 1 giá trị, 1 biến hoặc 1 sự kết hợp của hằng và biến,...
- Có thể sử dụng trong phép gán, cout hoặc 1 câu lệnh
 - `area = 2 * PI * radius;`
 - `cout << "border is: " << 2*(l+w);`
- Thứ tự thực hiện các toán tử trong 1 biểu thức như sau:
 - Phép phủ định đơn (unary negation), từ trái qua phải
 - *, /, % theo thứ tự từ trái qua phải
 - +, - theo thứ tự từ trái qua phải
- VD:

$$2 + 2 * \uparrow 2 - 2$$

Tính thứ 2 Tính thứ 1 Tính thứ 3

Biểu thức toán học (2)

- **Ví dụ minh họa biểu thức và giá trị của nó**

Table 3-2 Some Simple Expressions and Their Values

Expression	Value
5 + 2 * 4	13
10 / 2 - 3	2
8 + 12 * 2 - 4	28
4 + 17 % 2 - 1	4
6 - 3 * 2 + 7 - 1	6

- **Ví dụ minh họa kết hợp với dấu ngoặc đơn**

Table 3-4 More Simple Expressions and Their Values

Expression	Value
(5 + 2) * 4	28
10 / (5 - 3)	5
8 + 12 * (6 - 2)	56
(4 + 17) % 2 - 1	0
(6 - 3) * (2 + 7) / 3	9

Biểu thức đại số

- Một số ví dụ biểu thức đại số → biểu thức trong C++
 - $\text{Area} = l \cdot w \rightarrow \text{Area} = l * w;$
 - $\text{Area} = s^2 \rightarrow \text{Area} = \text{pow}(s, 2);$
 - $m = \frac{y_2 - y_1}{x_2 - x_1} \rightarrow m = (y_2 - y_1) / (x_2 - x_1);$

Table 3-5 Algebraic and C++ Multiplication Expressions

Algebraic Expression	Operation	C++ Equivalent
$6B$	6 times B	$6 * B$
$(3)(12)$	3 times 12	$3 * 12$
$4xy$	4 times x times y	$4 * x * y$

Chuyển đổi kiểu dữ liệu (1)

- Thao tác sẽ thực hiện giữa các toán hạng có cùng kiểu
- Nếu không cùng kiểu, C++ sẽ chuyển đổi thành cùng 1 kiểu
- Điều này có thể ảnh hưởng đến kết quả
- Phân cấp từ cao đến thấp kiểu dữ liệu:
 - long double
 - double
 - float
 - unsigned long
 - long
 - unsigned int
 - int

Chuyển đổi kiểu dữ liệu (2)

- Thao tác sẽ thực hiện giữa các toán hạng có cùng kiểu
- Nếu không cùng kiểu, C++ sẽ chuyển đổi thành cùng 1 kiểu
- Điều này có thể ảnh hưởng đến kết quả
- **Phân cấp kiểu dữ liệu từ cao đến thấp:**
 - long double, double, float, unsigned long, long, unsigned int, int
- **Một số quy tắc chuyển đổi:**
 - char, short, unsigned short → int
 - Khi thực hiện với các kiểu dữ liệu khác nhau thì kiểu thấp sẽ được nâng cấp lên kiểu cao hơn.
 - Khi sử dụng toán tử = thì kiểu của biểu thức bên phải sẽ chuyển đổi thành kiểu của biến bên trái.
- VD:
 - int **year** & float **interestRate**. Nếu **years * interestRate** thì trước khi thực hiện **year** sẽ nâng cấp lên thành float.
 - long int **area** & int **length, width**. Nếu **area = length * width** thì kết quả sẽ chuyển thành kiểu long int.

Tràn và dưới tràn – Overflow and underflow

- Xảy ra khi giá trị quá lớn (overflow) hoặc quá nhỏ (underflow) được gán cho một biến.
- Biến chứa các giá trị trong tập giá trị có thể
- Các hệ thống khác nhau sẽ có cách cảnh báo, báo lỗi, dừng/thực hiện chương trình với giá trị lỗi khác nhau.

Ép kiểu – Type Casting

- Sử dụng để ép kiểu 1 cách thủ công
- Cú pháp:
 - `static_cast<DataType>(Value)`
- Ví dụ:
 - `double number = 3.7;`
 - `int val;`
 - `val = static_cast<int>(number);`
- C chuẩn:
 - `cout << ch << " is " << (int)ch;`
- C chuẩn cũ:
 - `cout << ch << " is " << int(ch);`
- Cả 2 chuẩn đều chấp nhận dùng `static_cast`

Phép gán kết hợp và gán nhiều (1)

- Gán nhiều:
 - Dấu “=” có thể dùng để gán 1 giá trị cho nhiều biến:
 - $x = y = z = 5;$
 - Giá trị của $=$ là giá trị được gán
 - Thực hiện từ phải qua trái:
 - $x = (y = (z = 5));$

Giá trị là 5 Giá trị là 5 Giá trị là 5
- Gán kết hợp:
 - Ví dụ: `sum = sum + 1; // Thêm 1 vào biến sum`

Lệnh gán kết hợp và gán nhiều (2)

- Ví dụ về lệnh gán kết hợp:

Table 3-8 (Assume x = 6)

Statement	What It Does	Value of x After the Statement
x = x + 4;	Adds 4 to x	10
x = x - 3;	Subtracts 3 from x	3
x = x * 10;	Multiplies x by 10	60
x = x / 2;	Divides x by 2	3
x = x % 4	Makes x the remainder of x / 4	2

- Một số toán tử gán kết hợp:

Table 3-9

Operator	Example Usage	Equivalent to
+=	x += 5;	x = x + 5;
-=	y -= 2;	y = y - 2;
*=	z *= 10;	z = z * 10;
/=	a /= b;	a = a / b;
%=	c %= 3;	c = c % 3;

Định dạng đầu ra

- Có thể định dạng hiển thị dữ liệu đầu ra cho kiểu số, chuỗi:
 - Kích thước
 - Vị trí
 - Số chữ số
- Yêu cầu thư viện: iomanip
- Một số hàm:
 - setw(x): in ra giá trị với độ rộng x
 - fixed: hiển thị giá trị thực động thành thành thực cố định
 - setprecision(x): sẽ hiển thị x số sau dấu phẩy
 - showpoint: luôn in ra số thực động

Một số ví dụ về định dạng (1)

Program 3-13

```
1 // This program displays three rows of numbers.  
2 #include <iostream>  
3 #include <iomanip>      // Required for setw  
4 using namespace std;  
5  
6 int main()  
7 {  
8     int num1 = 2897, num2 = 5,      num3 = 837,  
9         num4 = 34,      num5 = 7,      num6 = 1623,  
10        num7 = 390,     num8 = 3456,    num9 = 12;  
11  
12    // Display the first row of numbers  
13    cout << setw(6) << num1 << setw(6)  
14        << num2 << setw(6) << num3 << endl;  
15  
16    // Display the second row of numbers  
17    cout << setw(6) << num4 << setw(6)  
18        << num5 << setw(6) << num6 << endl;  
19  
20    // Display the third row of numbers  
21    cout << setw(6) << num7 << setw(6)  
22        << num8 << setw(6) << num9 << endl;  
23    return 0;  
24 }
```

Program Output

2897	5	837
34	7	1623
390	3456	12

Một số ví dụ về định dạng (2)

Program 3-17

```
1 // This program asks for sales figures for 3 days. The total
2 // sales are calculated and displayed in a table.
3 #include <iostream>
4 #include <iomanip>
5 using namespace std;
6
7 int main()
8 {
9     double day1, day2, day3, total;
10
11    // Get the sales for each day.
12    cout << "Enter the sales for day 1: ";
13    cin >> day1;
14    cout << "Enter the sales for day 2: ";
15    cin >> day2;
16    cout << "Enter the sales for day 3: ";
17    cin >> day3;
18
19    // Calculate the total sales.
20    total = day1 + day2 + day3;
```

```
21
22     // Display the sales figures.
23     cout << "\nSales Figures\n";
24     cout << "-----\n";
25     cout << setprecision(2) << fixed;
26     cout << "Day 1: " << setw(8) << day1 << endl;
27     cout << "Day 2: " << setw(8) << day2 << endl;
28     cout << "Day 3: " << setw(8) << day3 << endl;
29     cout << "Total: " << setw(8) << total << endl;
30
31 }
```

Program Output with Example Input Shown in Bold

```
Enter the sales for day 1: 1321.87 [Enter]
Enter the sales for day 2: 1869.26 [Enter]
Enter the sales for day 3: 1403.77 [Enter]
```

```
Sales Figures
-----
Day 1: 1321.87
Day 2: 1869.26
Day 3: 1403.77
Total: 4594.90
```

Làm việc với đối tượng string và các kí tự

- Sử dụng cin và toán tử >> để nhập giá trị có thể gây ra:
 - Bỏ qua kí tự khoảng trắng nào: space, tab, breaks
 - Để giải quyết vấn đề này, có thể dùng hàm getline
- Để nhập 1 dòng chuỗi kí tự:
 - Sử dụng cin:
 - char ch;
 - cout << "Strike any key to continue";
 - cin >> ch;
 - Vấn đề: sẽ bỏ qua khoảng trắng, tabs, <CR>
 - Sử dụng cin.get():
 - cin.get(ch);
 - Sẽ đọc kí tự tiếp theo kể cả khoảng trắng

Một số ví dụ làm việc với string (1)

Program 3-19

```
1 // This program demonstrates using the getline function
2 // to read character data into a string object.
3 #include <iostream>
4 #include <string>
5 using namespace std;
6
7 int main()
8 {
9     string name;
10    string city;
11
12    cout << "Please enter your name: ";
13    getline(cin, name);
14    cout << "Enter the city you live in: ";
15    getline(cin, city);
16
17    cout << "Hello, " << name << endl;
18    cout << "You live in " << city << endl;
19    return 0;
20 }
```

Program Output with Example Input Shown in Bold

```
Please enter your name: Kate Smith [Enter]
Enter the city you live in: Raleigh [Enter]
Hello, Kate Smith
You live in Raleigh
```

Một số ví dụ làm việc với string (2)

Program 3-21

```
1 // This program demonstrates three ways
2 // to use cin.get() to pause a program.
3 #include <iostream>
4 using namespace std;
5
6 int main()
7 {
8     char ch;
9
10    cout << "This program has paused. Press Enter to continue.";
11    cin.get(ch);
12    cout << "It has paused a second time. Please press Enter again.";
13    ch = cin.get();
14    cout << "It has paused a third time. Please press Enter again.";
15    cin.get();
16    cout << "Thank you!";
17    return 0;
18 }
```

Program Output with Example Input Shown in Bold

This program has paused. Press Enter to continue. **[Enter]**
It has paused a second time. Please press Enter again. **[Enter]**
It has paused a third time. Please press Enter again. **[Enter]**
Thank you!

Một số vấn đề với string

- Sự trộn lẫn giữa `cin>>` và `cin.get()` trong cùng 1 chương trình có thể gây ra lỗi đầu vào.
- Để bỏ qua các kí tự không cần thiết trong bộ nhớ đệm bàn phím thì sử dụng `cin.ignore()`
 - `cin.ignore(); // skip next char`
 - `cin.ignore(10, '\n'); // skip the next 10 char or until a '\n'`
- Một số hàm trong thư viện string
 - Lấy độ dài của chuỗi:
 - `string state = "Texas";`
 - `int size = state.length();`
 - Nối nhiều chuỗi:
 - `greeting2 = greeting1 + name1;`
 - `greeting1 = greeting1 + name2;`

Hàm trong thư viện toán học (1)

- Yêu cầu đầu vào double và trả về double
- Làm việc với đầu vào double và trả về double
- Hàm thông thường:

sin	Sine
cos	Cosine
tan	Tangent
sqrt	Căn bậc 2
log	log
abs	Trị tuyệt đối (làm việc và trả về kiểu int)

Hàm trong thư viện toán học (2)

- Yêu cầu tệp header cstdlib
- rand(): trả về 1 số ngẫu nhiên (int) giữa 0 và số nguyên lớn nhất.
- srand(x): sinh số ngẫu nhiên ban đầu với x (unsigned int)

Theo dõi chương trình

- Hành động như một máy tính để thực hiện chương trình:
 - Từng bước thực hiện từng câu lệnh
 - Ghi lại giá trị các biến sau mỗi thay đổi, có thể dùng bảng
- Cách này rất hữu ích để phát hiện lỗi lôgic hoặc tính toán

Program 3-26 (with hand trace chart filled)

```
1 // This program asks for three numbers, then
2 // displays the average of the numbers.
3 #include <iostream>
4 using namespace std;

5 int main()
6 {
7     double num1, num2, num3, avg;
8     cout << "Enter the first number: ";
9     cin >> num1;
10    cout << "Enter the second number: ";
11    cin >> num2;
12    cout << "Enter the third number: ";
13    cin >> num3;
14    avg = num1 + num2 + num3 / 3;
15    cout << "The average is " << avg << endl;
16    return 0;
17 }
```

num1	num2	num3	avg
?	?	?	?
?	?	?	?
10	?	?	?
10	?	?	?
10	20	?	?
10	20	?	?
10	20	30	?
10	20	30	40
10	20	30	40

Bài tập thực hành

- Công ty ABC chuyên thiết kế các phẩm về thùng gỗ cho khách hàng.
 - Viết một chương trình để tính toán:
 - Số lượng khối gỗ (Volumn),
 - Chi phí (Cost),
 - Đơn giá (Price),
 - Lợi nhuận (Profit) mỗi sản phẩm.

Bảng tên hằng và tên biến

Table 3-14

Constant or Variable	Description
<code>cost_PER_CUBIC_FOOT</code>	A named constant, declared as a <code>double</code> and initialized with the value 0.23. This represents the cost to build a crate, per cubic foot.
<code>CHARGE_PER_CUBIC_FOOT</code>	A named constant, declared as a <code>double</code> and initialized with the value 0.5. This represents the amount charged for a crate, per cubic foot.
<code>length</code>	A <code>double</code> variable to hold the length of the crate, which is input by the user.
<code>width</code>	A <code>double</code> variable to hold the width of the crate, which is input by the user.
<code>height</code>	A <code>double</code> variable to hold the height of the crate, which is input by the user.
<code>volume</code>	A <code>double</code> variable to hold the volume of the crate. The value stored in this variable is calculated.
<code>cost</code>	A <code>double</code> variable to hold the cost of building the crate. The value stored in this variable is calculated.
<code>charge</code>	A <code>double</code> variable to hold the amount charged to the customer for the crate. The value stored in this variable is calculated.
<code>profit</code>	A <code>double</code> variable to hold the profit GCI makes from the crate. The value stored in this variable is calculated.

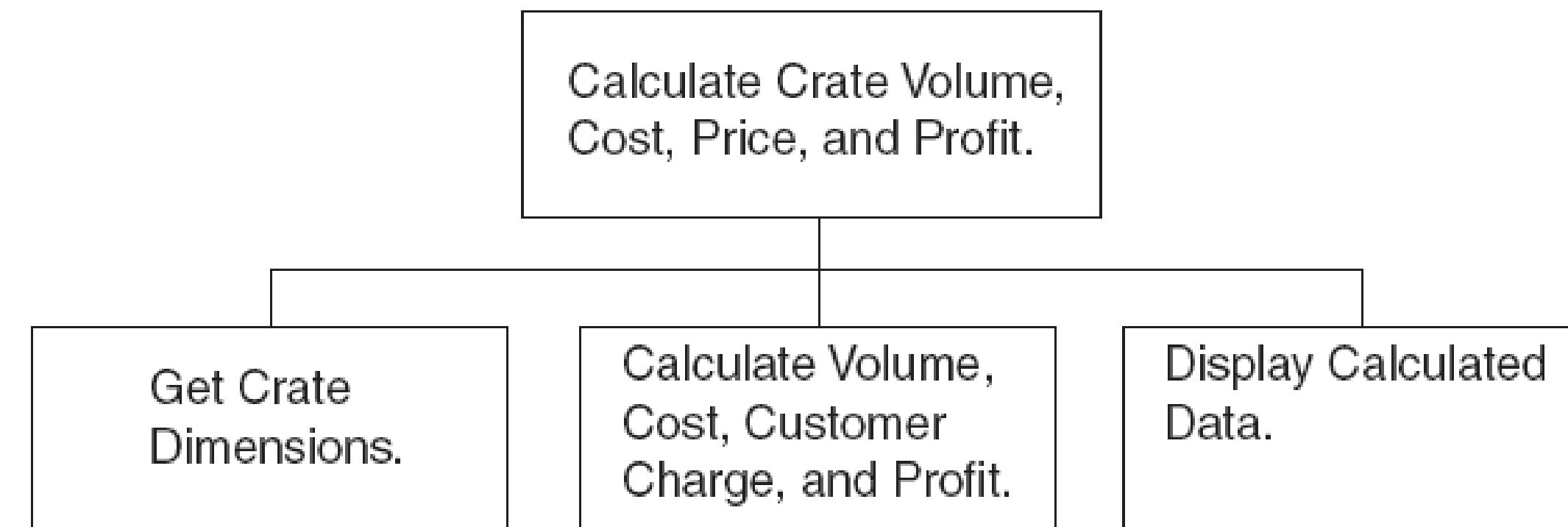
Thiết kế chương trình:

- Chương trình sẽ thực hiện theo bước sau:
 - Bước 1: Hỏi người dùng về kích thước thùng gỗ
 - Bước 2: Tính toán
 - Khối lượng thùng gỗ
 - Chi phí để tạo 1 thùng gỗ
 - Phí khách hàng
 - Lợi nhuận được tạo ra
 - Bước 3: Hiển thị dữ liệu tính toán trong bước 2.

Biểu đồ phân cấp chung

- Biểu đồ phân cấp chung:

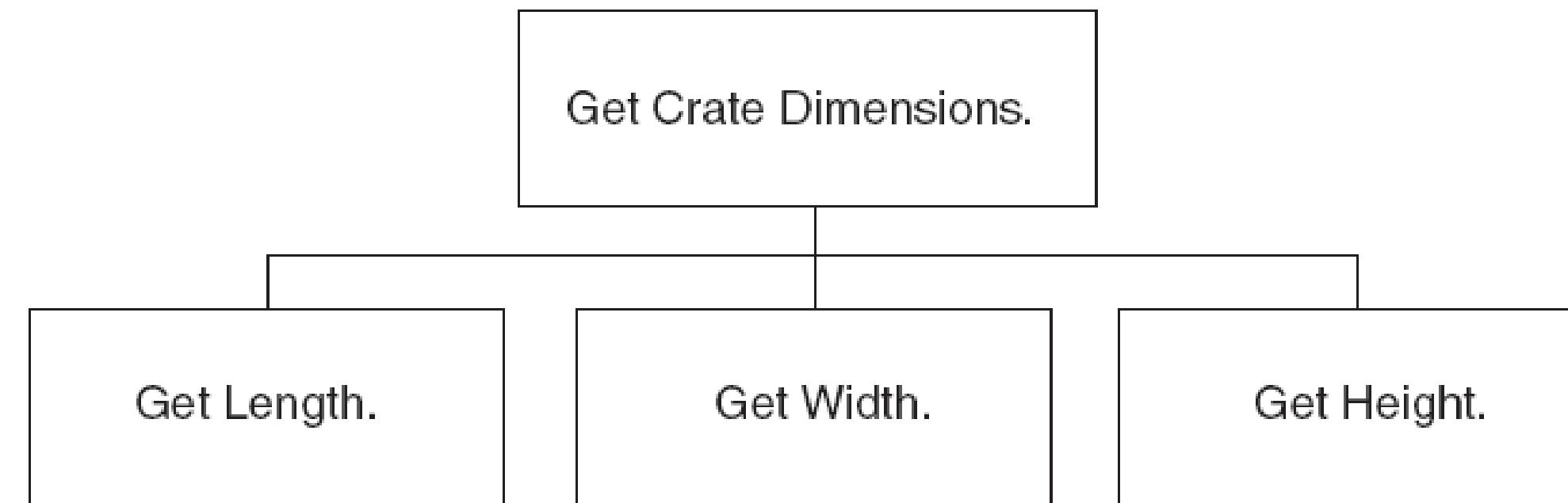
Figure 3-7



Biểu đồ thực hiện (1)

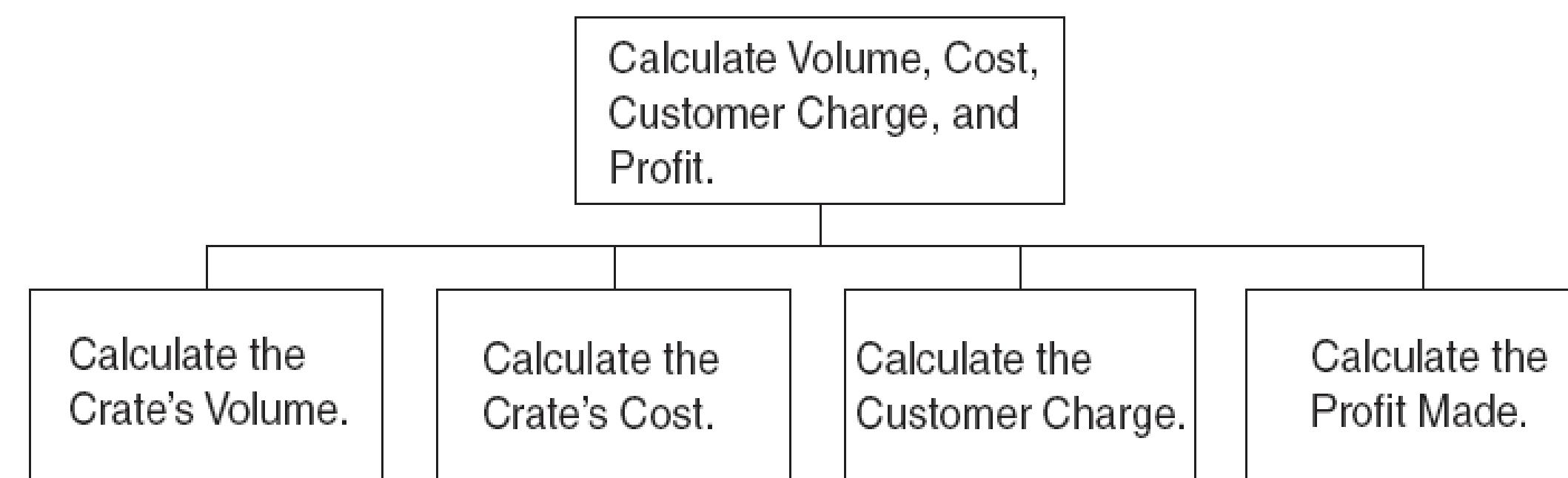
- **Lấy kích thước thùng gỗ:**

Figure 3-8



- **Tính toán: khối lượng, chi phí, phí khách hàng & lợi nhuận**

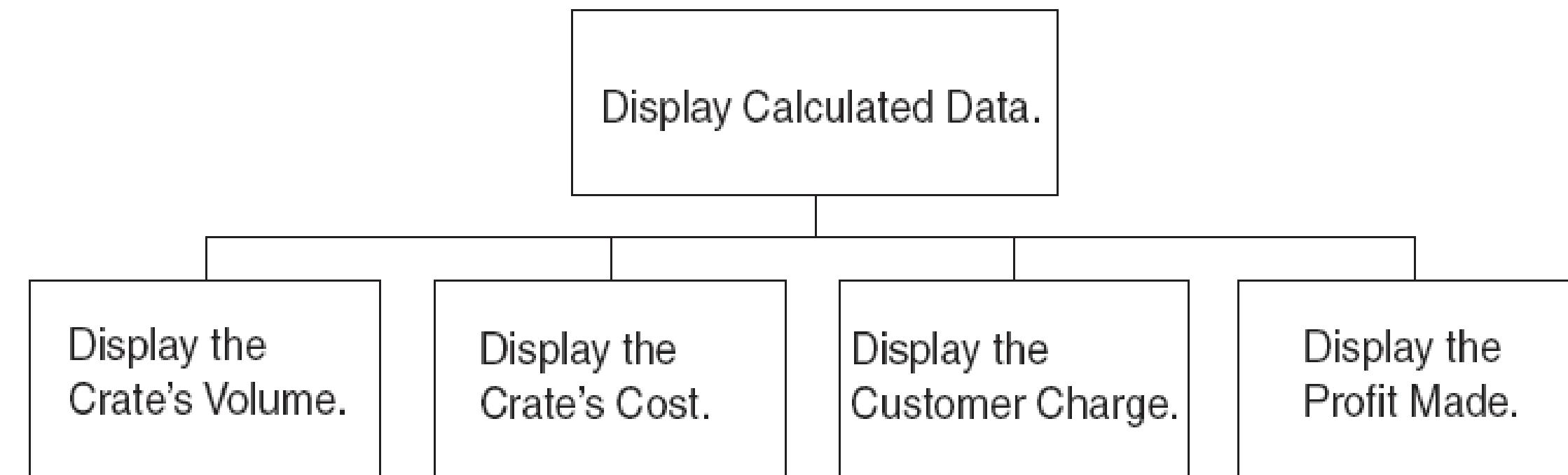
Figure 3-9



Biểu đồ thực hiện (2)

- **Hiển thị giá trị tính toán**

Figure 3-10



Giả mã - Psuedocode

- Hỏi người dùng nhập vào chiều dài thùng gỗ
- Hỏi người dùng nhập vào chiều rộng thùng gỗ
- Hỏi người dùng nhập vào chiều cao thùng gỗ
- Tính toán khối lượng thùng
- Tính toán chi phí để tạo ra 1 thùng
- Tính phí khách hàng cho thùng gỗ
- Tính toán lợi nhuận tạo ra từ 1 thùng
- Hiển thị khối lượng thùng gỗ
- Hiển thị chi phí tạo ra 1 thùng gỗ
- Hiển thị chi phí khách hàng
- Hiển thị lợi nhuận được tạo ra từ thùng gỗ

Tính toán

- Các công thức để tính toán như sau:
 - volume = length × width × height
 - cost = volume × 0.23
 - charge = volume × 0.5
 - profit = charge – cost

Chương trình (1)

Program 3-27

```
1 // This program is used by General Crates, Inc. to calculate
2 // the volume, cost, customer charge, and profit of a crate
3 // of any size. It calculates this data from user input, which
4 // consists of the dimensions of the crate.
5 #include <iostream>
6 #include <iomanip>
7 using namespace std;
8
9 int main()
10 {
11     // Constants for cost and amount charged
12     const double COST_PER_CUBIC_FOOT = 0.23;
13     const double CHARGE_PER_CUBIC_FOOT = 0.5;
14
15     // Variables
16     double length,      // The crate's length
17         width,        // The crate's width
18         height,       // The crate's height
19         volume,       // The volume of the crate
20         cost,         // The cost to build the crate
21         charge,       // The customer charge for the crate
22         profit;       // The profit made on the crate
23
24     // Set the desired output formatting for numbers.
25     cout << setprecision(2) << fixed << showpoint;
26 }
```

Continued...

Chương trình (2)

```
27 // Prompt the user for the crate's length, width, and height
28 cout << "Enter the dimensions of the crate (in feet):\n";
29 cout << "Length: ";
30 cin >> length;
31 cout << "Width: ";
32 cin >> width;
33 cout << "Height: ";
34 cin >> height;
35
36 // Calculate the crate's volume, the cost to produce it,
37 // the charge to the customer, and the profit.
38 volume = length * width * height;
39 cost = volume * COST_PER_CUBIC_FOOT;
40 charge = volume * CHARGE_PER_CUBIC_FOOT;
41 profit = charge - cost;
42
43 // Display the calculated data.
44 cout << "The volume of the crate is ";
45 cout << volume << " cubic feet.\n";
46 cout << "Cost to build: $" << cost << endl;
47 cout << "Charge to customer: $" << charge << endl;
48 cout << "Profit: $" << profit << endl;
49 return 0;
50 }
```

Continued...

Chương trình (3)

Program Output with Example Input Shown in Bold

Enter the dimensions of the crate (in feet):

Length: **10 [Enter]**

Width: **8 [Enter]**

Height: **4 [Enter]**

The volume of the crate is 320.00 cubic feet.

Cost to build: \$73.60

Charge to customer: \$160.00

Profit: \$86.40

Program Output with Different Example Input Shown in Bold

Enter the dimensions of the crate (in feet):

Length: **12.5 [Enter]**

Width: **10.5 [Enter]**

Height: **8 [Enter]**

The volume of the crate is 1050.00 cubic feet.

Cost to build: \$241.50

Charge to customer: \$525.00

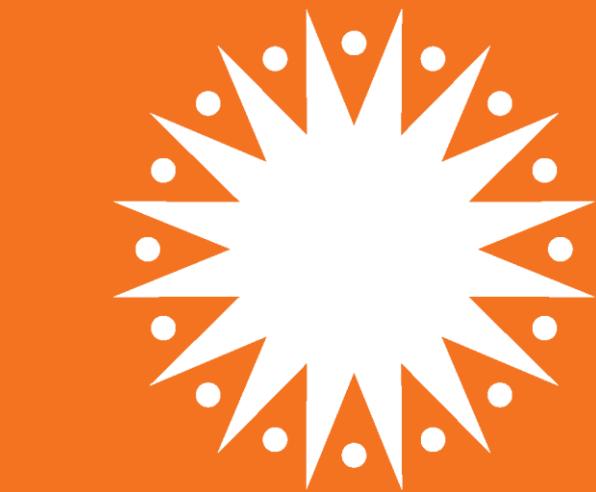
Profit: \$283.50

HỎI ĐÁP



Bài tập

- **Bài tập chương 2:**
 - Chapter2_Programming Challenges.pdf



ĐẠI NAM
UNIVERSITY

Trân trọng cảm ơn!

ĐẬU HẢI PHONG

Giảng viên

dauhaiphong@dainam.edu.vn

0912441435