

ĐẠI NAM
UNIVERSITY

Phần trình bày của:

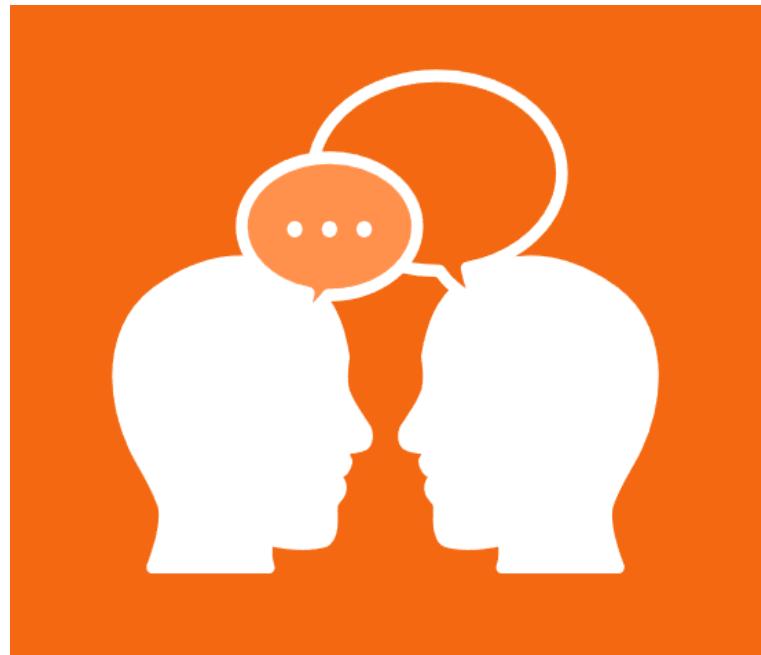
ĐẬU HẢI PHONG

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

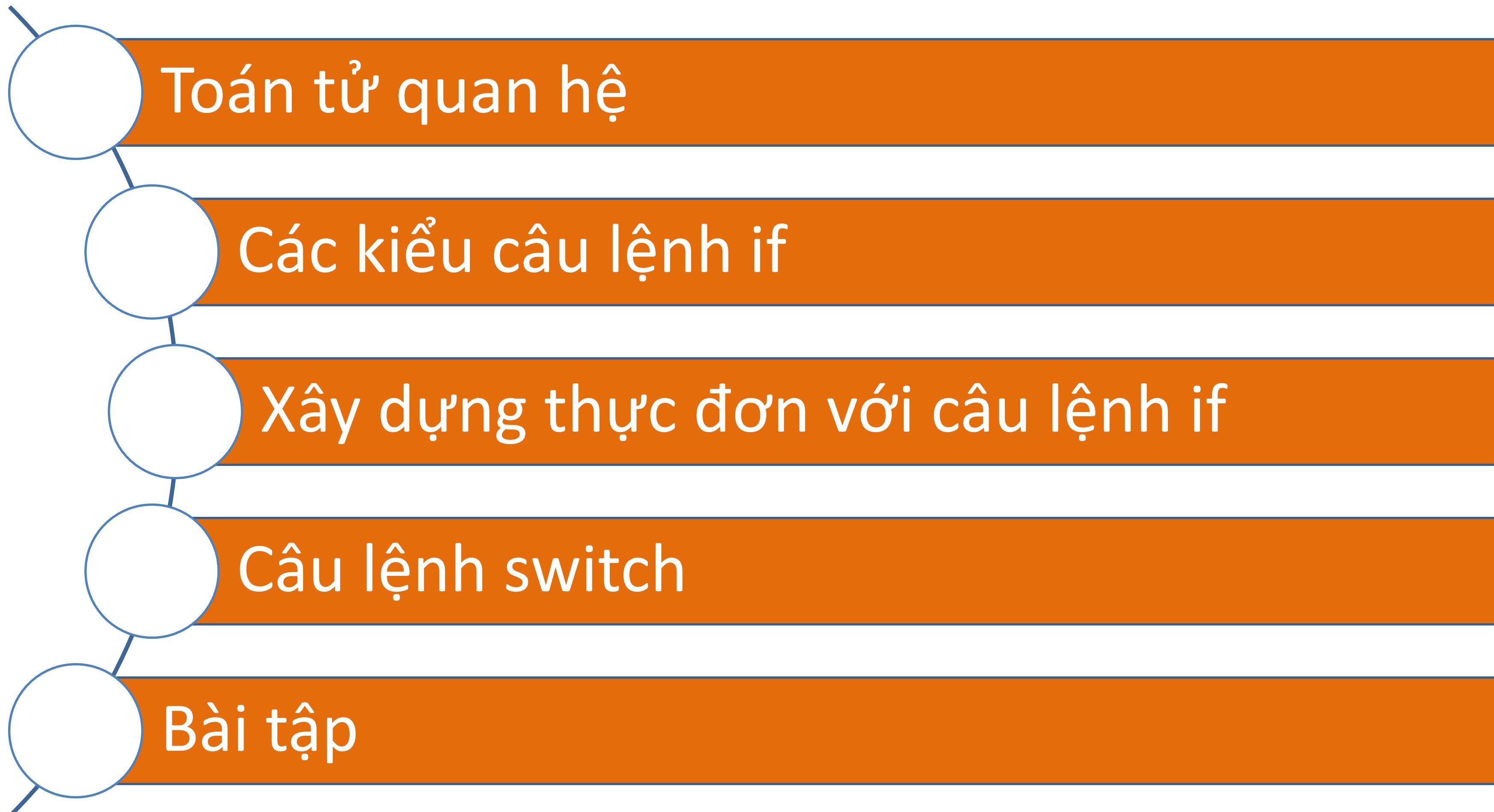


CHƯƠNG 3

CẤU TRÚC RẼ NHÁNH

NỘI DUNG

HỌC ĐỂ THAY ĐỔI



Toán tử quan hệ

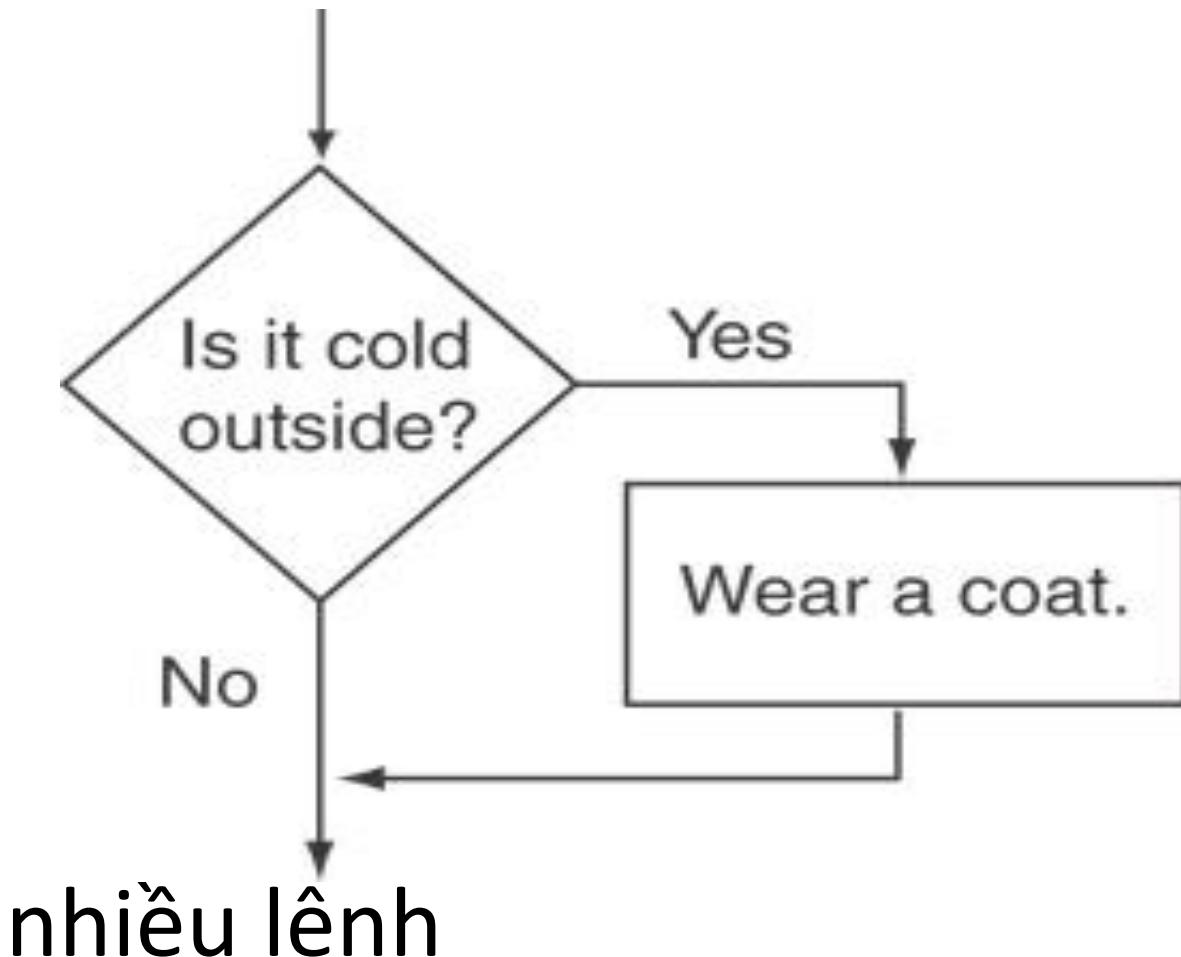
- Sử dụng so sánh các số hoặc kí tự (char) để xác định thứ tự quan hệ (lớn hơn, nhỏ hơn, bằng, lớn hơn hoặc bằng,...)
- Gồm các toán tử:
 - > lớn hơn
 - < nhỏ hơn
 - \geq lớn hơn hoặc bằng
 - \leq nhỏ hơn hoặc bằng
 - \equiv **bằng**
 - \neq khác
- Biểu thức quan hệ: trả về true hoặc false
 - VD1: $12 < 5$ là false; $7 \geq 7$ là true
 - VD2: $result = x \leq y;$ // bằng 0 ~ false, 1 ~ true

Câu lệnh if

- Cho phép câu lệnh hoặc nhóm các câu lệnh được thực hiện theo một số điều kiện nhất định.
- Hoạt động theo cách đánh giá tình huống:
 - Nếu trời mưa thì mang ô
 - Nếu trời bên ngoài lạnh thì mặc áo khoác.
- Cú pháp:

```
if (biểu thức)
  Câu lệnh;
  ...
}
```
- Chú ý:
 - Không có dấu chấm phẩy ; sau (biểu thức)
 - 0 là **false** và khác 0 luôn là giá trị **true**

```
if (biểu thức) {
  Câu lệnh 1;
  Câu lệnh n;
}
```



```
if (SoBai>=20) {
  cout<<"NGÔI SAO LẬP TRÌNH TUẦN 2";
}
```

Ví dụ câu lệnh if

Program 4-2

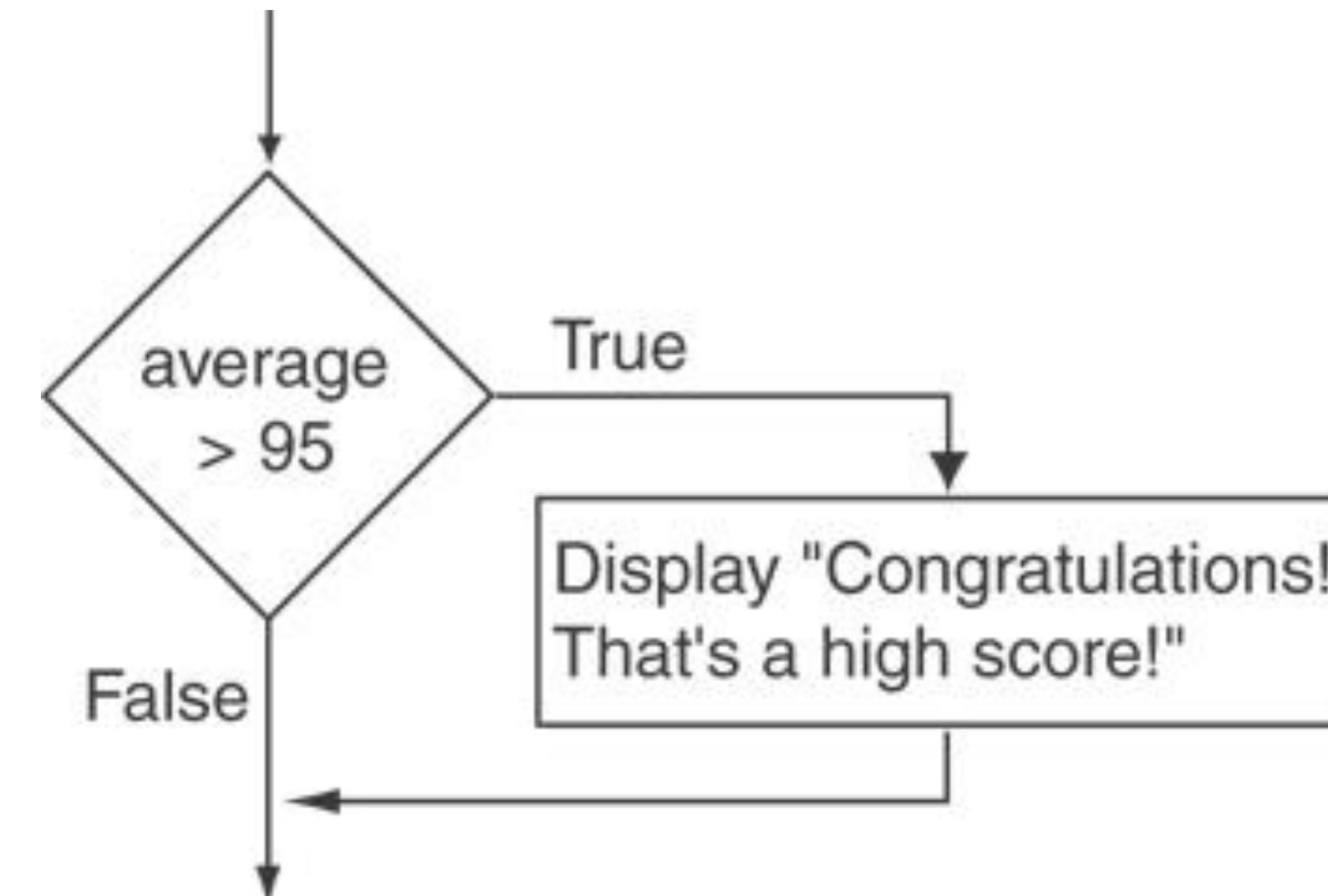
```
1 // This program averages three test scores
2 #include <iostream>
3 #include <iomanip>
4 using namespace std;
5
6 int main()
7 {
8     int score1, score2, score3; // To hold three test scores
9     double average;           // To hold the average score
10
11    // Get the three test scores.
12    cout << "Enter 3 test scores and I will average them: ";
13    cin >> score1 >> score2 >> score3;
14
15    // Calculate and display the average score.
16    average = (score1 + score2 + score3) / 3.0;
17    cout << fixed << showpoint << setprecision(1);
18    cout << "Your average is " << average << endl;
19
20    // If the average is greater than 95, congratulate the user.
21    if (average > 95)
22        cout << "Congratulations! That's a high score!\n";
23    return 0;
24 }
```

Program Output with Example Input Shown in Bold

Enter 3 test scores and I will average them: **80 90 70 [Enter]**
Your average is 80.0

Program Output with Other Example Input Shown in Bold

Enter 3 test scores and I will average them: **100 100 100 [Enter]**
Your average is 100.0
Congratulations! That's a high score!



Câu lệnh if/else

- Cú pháp:

if (biểu thức)

Câu lệnh 1; // hoặc khối lệnh có thêm {}

else

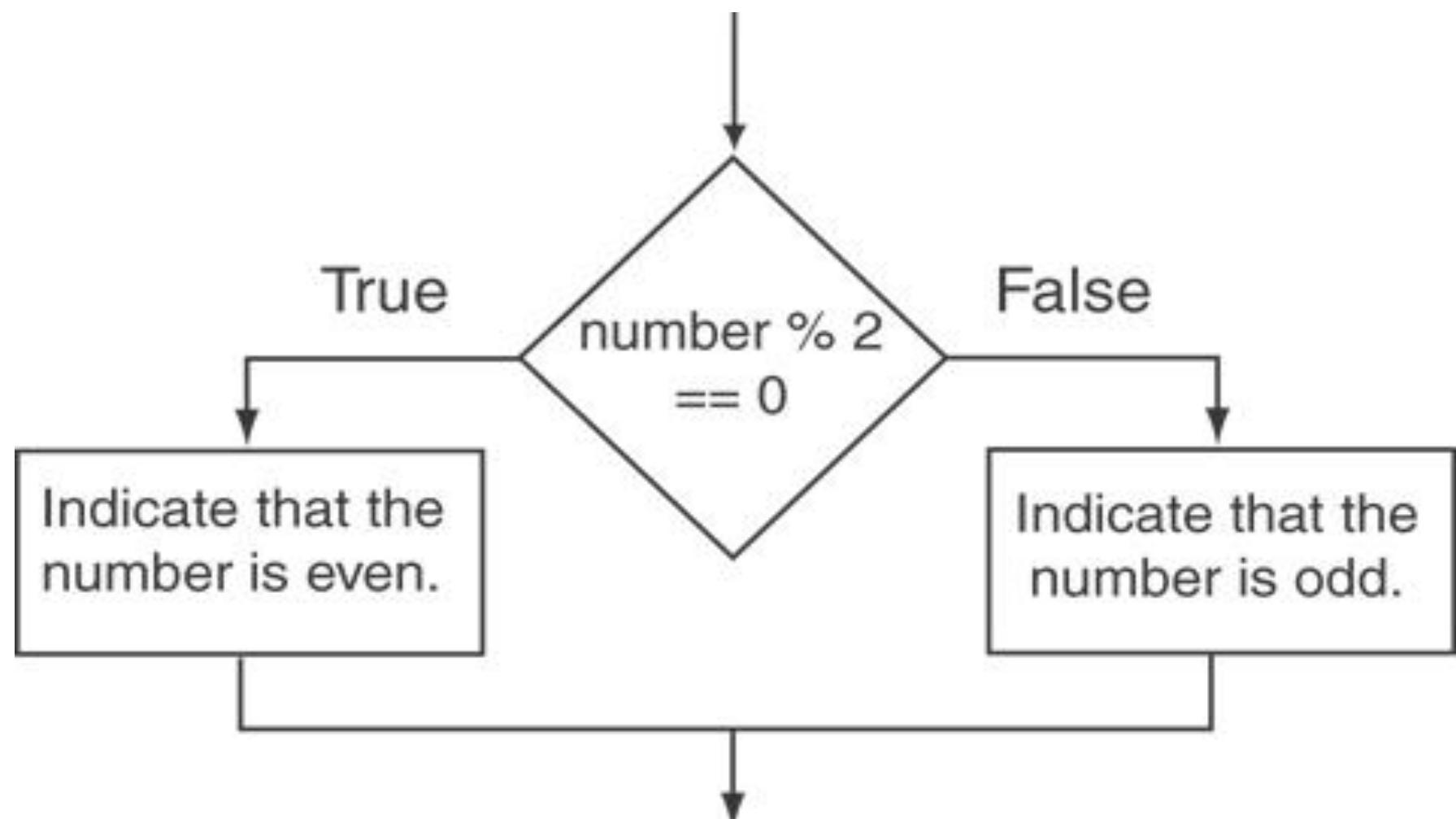
Câu lệnh 2; // hoặc khối lệnh có thêm {}

- Nếu biểu thức là ĐÚNG thì thực hiện **Câu lệnh 1** và bỏ qua **Câu lệnh 2**.
- Nếu biểu thức là SAI thì **Câu lệnh 1** được bỏ qua và thực hiện **Câu lệnh 2**.

Câu lệnh if/else

Program 4-8

```
1 // This program uses the modulus operator to determine
2 // if a number is odd or even. If the number is evenly divisible
3 // by 2, it is an even number. A remainder indicates it is odd.
4 #include <iostream>
5 using namespace std;
6
7 int main()
8 {
9     int number;
10
11    cout << "Enter an integer and I will tell you if it\n";
12    cout << "is odd or even. ";
13    cin >> number;
14    if (number % 2 == 0)
15        cout << number << " is even.\n";
16    else
17        cout << number << " is odd.\n";
18    return 0;
19 }
```



Program Output with Example Input Shown in Bold

Enter an integer and I will tell you if it
is odd or even. **17 [Enter]**
17 is odd.

Câu lệnh if/else

Program 4-9

```
1 // This program asks the user for two numbers, num1 and num2.  
2 // num1 is divided by num2 and the result is displayed.  
3 // Before the division operation, however, num2 is tested  
4 // for the value 0. If it contains 0, the division does not  
5 // take place.  
6 #include <iostream>  
7 using namespace std;  
8  
9 int main()  
10 {  
11     double num1, num2, quotient;  
12  
13     // Get the first number.  
14     cout << "Enter a number: ";  
15     cin >> num1;  
16  
17     // Get the second number.  
18     cout << "Enter another number: ";  
19     cin >> num2;  
20  
21     // If num2 is not zero, perform the division.  
22     if (num2 == 0)  
23     {  
24         cout << "Division by zero is not possible.\n";  
25         cout << "Please run the program again and enter\n";  
26         cout << "a number other than zero.\n";  
27     }  
28     else  
29     {  
30         quotient = num1 / num2;  
31         cout << "The quotient of " << num1 << " divided by "  
32         cout << num2 << " is " << quotient << ".\n";  
33     }  
34     return 0;  
35 }
```

Program Output with Example Input Shown in Bold

(When the user enters 0 for num2)

Enter a number: **10** [Enter]

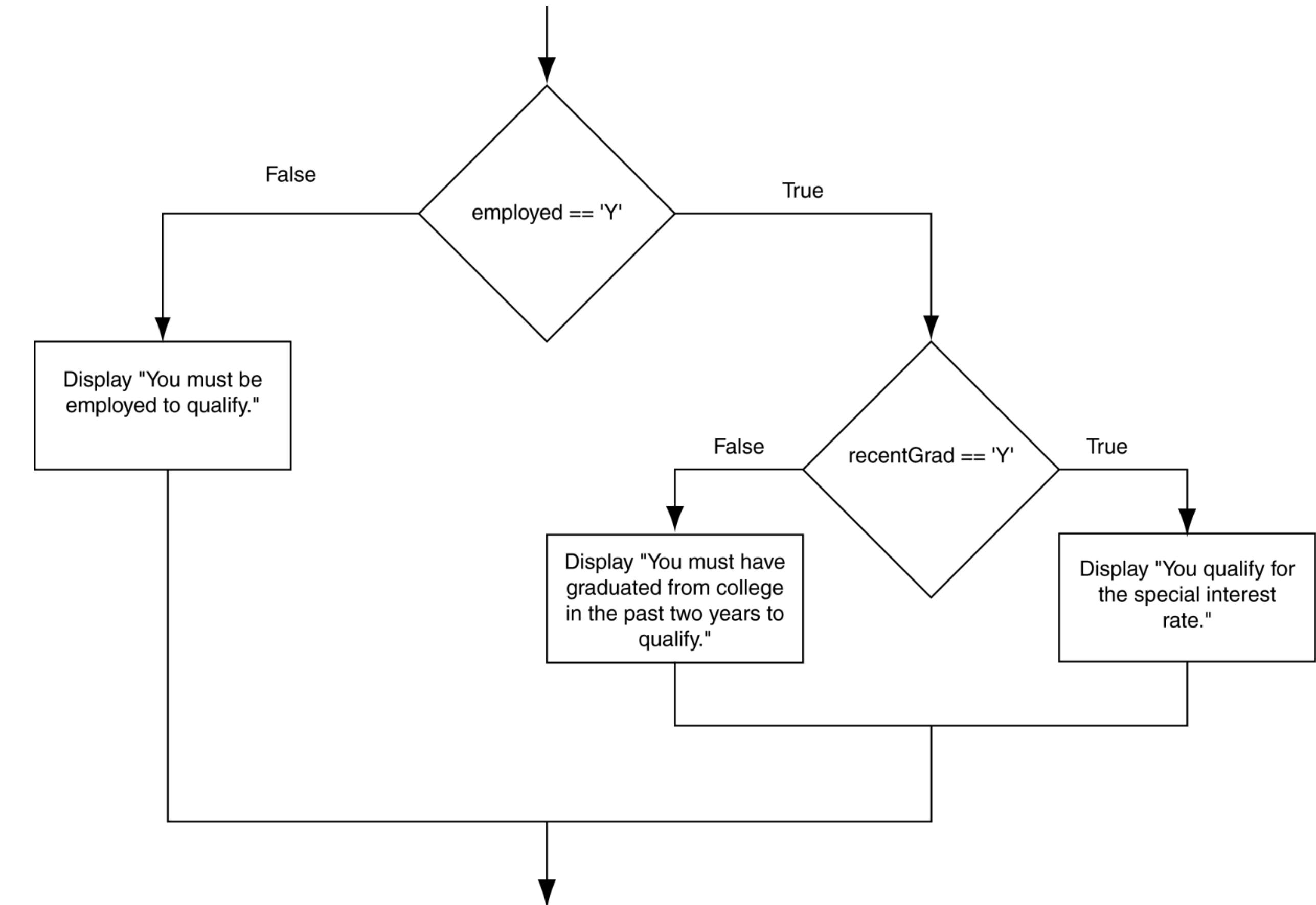
Enter another number: **0** [Enter]

Division by zero is not possible.

Please run the program again and enter
a number other than zero.

Câu lệnh *if* lồng nhau

- Để kiểm tra nhiều hơn một điều kiện,
- Một câu lệnh *if* có thể được lồng bên trong một câu lệnh *if* khác.



Câu lệnh *if* lồng nhau

- Để kiểm tra nhiều hơn một điều kiện,
- Một câu lệnh *if* có thể được lồng bên trong một câu lệnh *if* khác.

```
• 20 // Determine the user's loan qualifications.  
21 if (employed == 'Y')  
22 {  
23     if (recentGrad == 'Y') //Nested if  
24     {  
25         cout << "You qualify for the special "  
26         cout << "interest rate.\n";  
27     }  
28 }
```

Câu lệnh *if* lồng nhau

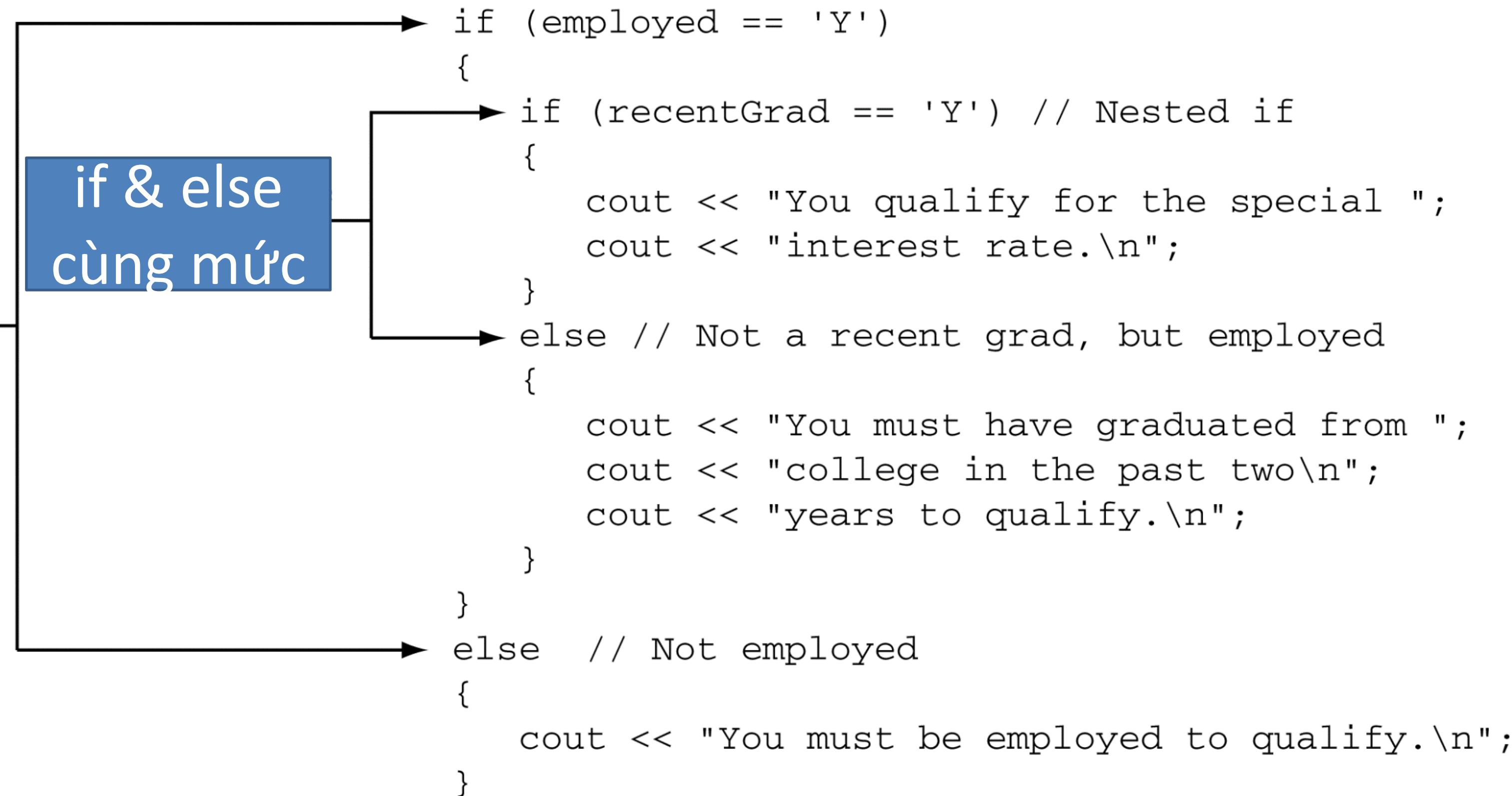
- Ví dụ: Program 4-11

```
20 // Determine the user's loan qualifications.  
21 if (employed == 'Y')  
22 {  
23     if (recentGrad == 'Y') // Nested if  
24     {  
25         cout << "You qualify for the special ";  
26         cout << "interest rate.\n";  
27     }  
28     else // Not a recent grad, but employed  
29     {  
30         cout << "You must have graduated from ";  
31         cout << "college in the past two\n";  
32         cout << "years to qualify.\n";  
33     }  
34 }  
35 else // Not employed  
36 {  
37     cout << "You must be employed to qualify.\n";  
38 }
```

Câu lệnh *if* lồng nhau

- Chú ý đẩy tab (lùi đầu dòng) *if* & *else* tương ứng

if & else
cùng mức



```
if (employed == 'Y')
{
    if (recentGrad == 'Y') // Nested if
    {
        cout << "You qualify for the special ";
        cout << "interest rate.\n";
    }
    else // Not a recent grad, but employed
    {
        cout << "You must have graduated from ";
        cout << "college in the past two\n";
        cout << "years to qualify.\n";
    }
}
else // Not employed
{
    cout << "You must be employed to qualify.\n";
}
```

The diagram illustrates nested if statements. A blue box labeled "if & else cùng mức" (at the same level) is positioned above the inner if block. Arrows point from the "if" and "else" keywords of the inner block to their corresponding counterparts in the outer block, demonstrating that they are at the same indentation level.

Câu lệnh *if / else if*

- Cú pháp:

if (biểu thức 1)

 Câu lệnh 1; // hoặc khối lệnh

else if (biểu thức 2)

 Câu lệnh 2; // hoặc khối lệnh

...

.

else if (biểu thức 3)

 Câu lệnh 3; // hoặc khối lệnh

- Sẽ kiểm tra 1 chuỗi các điều kiện

- Thường dùng cho đơn giản hơn so với *if/else* lồng nhau.

```
21     // Determine the letter grade.  
22     if (testScore >= A_SCORE)  
23         cout << "Your grade is A.\n";  
24     else if (testScore >= B_SCORE)  
25         cout << "Your grade is B.\n";  
26     else if (testScore >= C_SCORE)  
27         cout << "Your grade is C.\n";  
28     else if (testScore >= D_SCORE)  
29         cout << "Your grade is D.\n";  
30     else  
31         cout << "Your grade is F.\n";
```

Câu lệnh *if / else if*

```
if (expression_1)
{
    statement
    statement
    etc.
}

else if (expression_2)
{
    statement
    statement
    etc.
}

Insert as many else if clauses as necessary

else
{
    statement
    statement
    etc.
}
```

If *expression_1* is true these statements are executed, and the rest of the structure is ignored.

Otherwise, if *expression_2* is true these statements are executed, and the rest of the structure is ignored.

These statements are executed if none of the expressions above are true.

Biến cờ - Flags

- Biến cờ dùng để đánh dấu sự tồn tại của một điều kiện
- Có kiểu bool hoặc kiểu nguyên
- Thường được gán một giá trị ban đầu
- Ví dụ: chương trình cần xác định người bán có đáp ứng được hạn ngạch không? Thay vì in ra màn hình thông báo “Có đáp ứng được hạn ngạch!”
- Đoạn chương trình:

```
bool salesQuotaMet = false;  
if (sales >= QUOTA_AMOUNT)  
    salesQuotaMet = true;  
else  
    salesQuotaMet = false;
```

Toán tử lôgic

- Sử dụng để tạo biểu thức logic từ các biểu thức logic khác
- Gồm các toán tử:
 - `&&` ($BT1 \&\& BT2$) Đúng nếu cả BT1 và BT2 cùng đúng
 - `||` ($BT1 || BT2$) Đúng nếu cả BT1 hoặc BT2 đúng
 - `!` $!(BT1)$ Phủ định của BT1
- Ví dụ: `int x = 12, y = 5, z = -4;`

$(x > y) \&\& (y > z)$	true
$(x > y) \&\& (z > y)$	false
$(x \leq z) (y == z)$	false
$(x \leq z) (y != z)$	true
$! (x \geq z)$	false

Toán tử lôgic

- Ví dụ: đoạn mã Program 4-15, 4-16, 4-17

```
23 // Determine the user's loan qualifications.
24 if (!(income >= MIN_INCOME || years > MIN_YEARS))
25 {
26     cout << "You must earn at least $"
27             << MIN_INCOME << " or have been "
28             << "employed more than " << MIN_YEARS
29             << " years.\n";
30 }
31 else
32     cout << "You qualify.\n";
--           -----
32 }           << "past two years to qualify.\n";
32 }
```

Toán tử logic

- Một số chú ý:
 - Thứ tự ưu tiên trong toán tử logic lần lượt là: ! → && → ||
 - Toán tử ! cho 1 biểu thức nên để ngoài ngoặc
 - Phân tích ví dụ sau:
 - $!(x > 2)$
 - $!x > 2$
 - Nên dùng dấu () để xác định biểu thức nào cần thực hiện trước.
 - VD:
 - $(a < b) || ((y == z) \&\& (m > j))$

Kiểm tra phạm vi của 1 số bằng toán tử logic

- Ví dụ 1: để xác định x nằm NGOÀI phạm vi 20 đến 40

```
if (x < 20 || x > 40)  
  
    cout << x << " is outside the acceptable range.\n";
```

- Ví dụ 2: để xác định x nằm TRONG phạm vi 20 đến 40

```
if (x < 20 && x > 40)  
  
    cout << x << " is outside the acceptable range.\n";
```

- Chú ý: **không** viết như sau $20 < x < 40$

Thực hành: viết thực đơn lựa chọn

- Sử dụng câu lệnh if/else hay if/else lồng nhau để tạo chương trình theo kiểu thực đơn.
 - Chương trình cho phép người dùng lựa chọn hành động từ danh sách hành động có sẵn.
 - Thực đơn: hiển thị danh sách lựa chọn trên màn hình
 - Cách tổ chức như sau:
 - Hiện thị các số hoặc kí tự tương ứng với hành động
 - Nhắc người dùng lựa chọn
 - Kiểm tra người dùng lựa chọn bằng biểu thức
 - Ví dụ thực đơn
 - Minh họa bằng chương trình 4-18
1. Add a name to the list.
2. Remove a name from the list.
3. Change a name in the list.
4. Print the list.
5. Quit the program.

Xác thực người dùng nhập vào

- Khi nhập giá trị đầu vào, có thể người dùng nhập giá trị không chính xác.
- Nên cần kiểm tra giá trị nhập vào
- Có thể xác thực việc sau:
 - Kiểm tra giá trị trong khoảng.
 - VD: 1 ngày chỉ 24 tiếng, nhập nhiều hơn sẽ báo lỗi
 - Kiểm tra sự bất hợp lý.
 - VD: 1 người không thể làm việc 24 tiếng, mặc dù có thể
 - Kiểm tra lựa chọn thực đơn.
 - VD: Nếu lựa chọn chức năng nào đó của thực đơn
 - Kiểm tra chia không
 - VD: Nếu số chia là 0 thì báo lỗi
 - ...
- Minh họa pr4-19

So sánh kí tự và xâu kí tự

- Kí tự so sánh theo giá trị của kí tự trong bảng mã ASCII
 - VD1: 'A' < 'B', vì trong bảng mã ASCII 'A'(65) và 'B'(66)
 - VD2: '1' < '2', vì trong bảng mã ASCII '1'(49) và '2'(50)
 - Kí tự thường có giá trị trong bảng mã ASCII cao hơn chữ Hoa
 - VD: 'a' > 'Z'
 - Minh họa **pr4-20**
- Xâu kí tự cũng so sánh theo bảng mã ASCII
 - string name1 = "Mary";
 - string name2 = "Mark";
 - name1 > name2 // true
 - name1 <= name2 // false
 - name1 != name2 // true
 - name1 < "Mary Jane" // true
 - Minh họa **pr4-21**

Toán tử điều kiện

- Có thể sử dụng để tạo câu lệnh *if/else* ngắn gọn
- Cú pháp: `expr1 ? expr2 : expr3;`

Điều kiện này sẽ được kiểm tra đầu tiên

Biểu thức này được thực hiện thứ 2 nếu `expr1` là ĐÚNG

Biểu thức này được thực hiện thứ 3 nếu `expr1` là SAI

- VD: `x<0 ? y=10 : z=20;`
- Minh họa pr4-22

Câu lệnh switch

- Câu lệnh switch cho phép xác định giá trị của 1 biến hay biểu thức sẽ phân nhánh.
- Trong một số trường hợp có thể thay thế if/else
- Cú pháp:

```
switch (expression) //integer
{
    case exp1: statement1;
    case exp2: statement2;
    ...
    case expn: statementn;
    default: statementn+1;
}
```

Program 4-23

```
1 // The switch statement in this program tells the user something
2 // he or she already knows: the data just entered!
3 #include <iostream>
4 using namespace std;
5
6 int main()
7 {
8     char choice;
9
10    cout << "Enter A, B, or C: ";
11    cin >> choice;
12    switch (choice)
13    {
14        case 'A': cout << "You entered A.\n";
15                    break;
16        case 'B': cout << "You entered B.\n";
17                    break;
18        case 'C': cout << "You entered C.\n";
19                    break;
20        default: cout << "You did not enter A, B, or C!\n";
21    }
22    return 0;
23 }
```

Program Output with Example Input Shown in Bold

Enter A, B, or C: **B** [Enter]
You entered B.

Program Output with Example Input Shown in Bold

Enter A, B, or C: **F** [Enter]
You did not enter A, B, or C!

Câu lệnh switch

- Câu lệnh switch yêu cầu:
 - Biểu thức (expression) phải là 1 biến nguyên (interger) hoặc 1 biểu thức trả về giá trị nguyên.
 - exp1 đến expn phải là hằng số hoặc số nguyên và là duy nhất trong câu lệnh switch.
 - VD: case x < 5: hoặc case n==0: **là sai**
 - **default** là giá trị mặc định nếu không có giá nào thỏa mãn với switch.

Câu lệnh switch

- Cách câu lệnh switch thực hiện:
 - Giá trị của *expression* so sánh với các giá trị exp1 đến expn
 - Nếu *expression* khớp với giá trị expi thì chương trình sẽ nhánh để thực hiện các câu lệnh của nhánh đó.
 - Nếu *expression* không khớp với giá trị expi nào thì sẽ thực hiện các câu lệnh trong default.
- Chú ý:
 - Sử dụng break để thoát khỏi switch
 - Nếu gặp break thì chương trình sẽ bỏ qua các câu lệnh phía dưới

Câu lệnh switch

Program 4-25

```
1 // This program is carefully constructed to use the "fall through"
2 // feature of the switch statement.
3 #include <iostream>
4 using namespace std;
5
6 int main()
7 {
8     int modelNum; // Model number
9
10    // Get a model number from the user.
11    cout << "Our TVs come in three models:\n";
12    cout << "The 100, 200, and 300. Which do you want? ";
13    cin >> modelNum;
14
15    // Display the model's features.
16    cout << "That model has the following features:\n";
17    switch (modelNum)
18    {
19        case 300: cout << "\tPicture-in-a-picture.\n";
20        case 200: cout << "\tStereo sound.\n";
21        case 100: cout << "\tRemote control.\n";
22            break;
23        default: cout << "You can only choose the 100,";
24                cout << "200, or 300.\n";
25    }
26    return 0;
27 }
```

Program Output with Example Input Shown in Bold

Our TVs come in three models:

The 100, 200, and 300. Which do you want? **100 [Enter]**

That model has the following features:

 Remote control.

Program Output with Example Input Shown in Bold

Our TVs come in three models:

The 100, 200, and 300. Which do you want? **200 [Enter]**

That model has the following features:

 Stereo sound.

 Remote control.

Program Output with Example Input Shown in Bold

Our TVs come in three models:

The 100, 200, and 300. Which do you want? **300 [Enter]**

That model has the following features:

 Picture-in-a-picture.

 Stereo sound.

 Remote control.

Program Output with Example Input Shown in Bold

Our TVs come in three models:

The 100, 200, and 300. Which do you want? **500 [Enter]**

That model has the following features:

You can only choose the 100, 200, or 300.

Sử dụng switch với thực đơn

- Câu lệnh switch là lựa chọn thích hợp với chương trình sử dụng thực đơn dạng như sau:
 - Hiển thị thực đơn
 - Đưa cho người dùng các lựa chọn
 - Người dùng nhập vào 1 giá trị trong câu lệnh switch
 - Kiểm tra và thực hiện nhóm lệnh tương ứng với giá trị nhập vào

Khối lệnh và phạm vi biến

- Phạm vi biến được giới hạn trong phạm vi khối lệnh nó định nghĩa.
- Có thể khai báo cùng 1 tên biến nếu khác khôi.

Program 4-30

```
1 // This program uses two variables with the name number.
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     // Define a variable named number.
8     int number;
9
10    cout << "Enter a number greater than 0: ";
11    cin >> number;
12    if (number > 0)
13    {
14        int number; // Another variable named number.
15        cout << "Now enter another number: ";
16        cin >> number;
17        cout << "The second number you entered was "
18             << number << endl;
19    }
20    cout << "Your first number was " << number << endl;
21    return 0;
22 }
```

Program Output with Example Input Shown in Bold

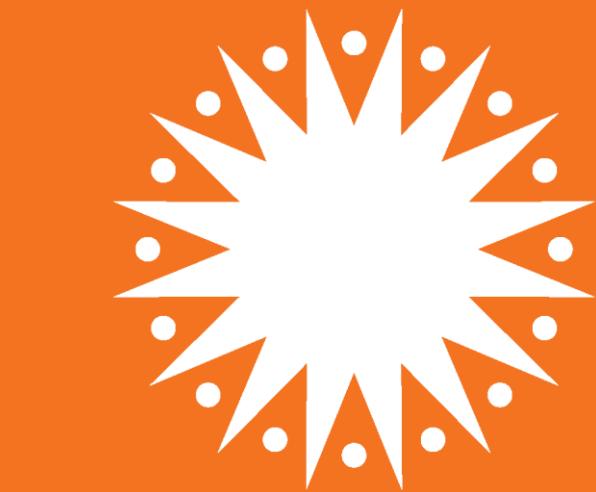
```
Enter a number greater than 0: 2 [Enter]
Now enter another number: 7 [Enter]
The second number you entered was 7
Your first number was 2
```

HỎI ĐÁP



Bài tập

- **Bài tập chương 3:**
 - Chapter3_Programming Challenges.pdf



ĐẠI NAM
UNIVERSITY

Trân trọng cảm ơn!

ĐẬU HẢI PHONG

Giảng viên

dauhaiphong@dainam.edu.vn

0912441435