Tests

CSE 232 - Dr. Josh Nahum

Reading:

Section 1.8

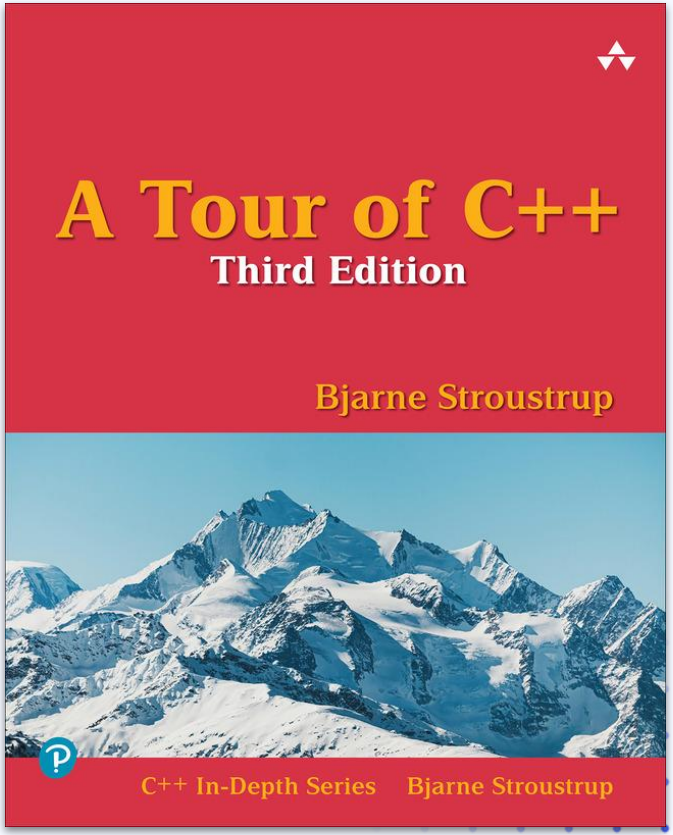

Table of contents

OO

if / else

flow control bodies

single statement body

```
if (x == 6)
  std::cout << "x is six";
```

No need for braces if body consists of just one statement

multi-statement body

```
if (x == 6) {
  std::cout << "x is six";
  ++x;
}
```

Braces allow the body of a flow control statement to consist of zero or more statements

else statements

single statement body

```
if ( x == 6)
  std::cout << "x is six";
else
  std::cout << "not six";
```

multi-statement body

```
if (x == 6) {
  std::cout << "x is six";
  ++x;
} else {
  std::cout << "not six"; --x;
}
```

conditional expressions

"true" values

```
if (x)
while (x)
for (...; x; ...)
```

All of the above check if x has a "true" value. Any non-zero value is considered true.

Both equivalent

```
if (x)
if (x != 0)

if (!x)
if (x == 0)

'\0', 0,  nullptr
```

These are all the same

01

break / continue

break

```
for (int x = 0; x < 5; ++x) {
  if (x % 3 == 0) {
    cout << "x is evenly divisible by 3"
         << endl;
    break;
  }
  cout << "x is " << x << endl;
}
```

Prints out:

```
x is evenly divisible by 3
```

continue

```
for (int x = 0; x < 5; ++x) {
  if (x % 3 == 0) {
    cout << "x is evenly divisible by 3"
         << endl;
    continue;
  }
  cout << "x is " << x << endl;
}
```

Prints out:

```
x is evenly divisible by 3
x is 1
x is 2
x is evenly divisible by 3
x is 4
```

02

for loop details

for loop (while loop)

```
for (init_statement; condition; iteration_expression)
  statement
```

Mostly equivalent to:

```
init_statement;
while (condition) {
  statement;
  iteration_expression;
}
```

for loop (optional parts)

```
for (init_statement; condition; iteration_expression)
  statement
```

The init_statement, condition, and iteration_expression are all optional. An empty condition is always true.

```
for (;;) { // Infinite loop unless there is a break statement.
  ...
}
```

for loop (scope)

```
for (init_statement; condition; iteration_expression)
  statement
```

Variables declared in the init_statement are in scope until the end of the for loop.

As mentioned previously, a continue statement causes the for loop statement to end, with the iteration_expression being executed next.

03

other flow control

Other flow control

Do-while loops

The body is executed before the condition is checked

Conditional operator

An operator that allows flow control within an expression

Goto

A statement that allows jumping to a label

The above are optional content that is not required for 232.

Attribution

Please ask questions via Piazza

Dr. Joshua Nahum
www.nahum.us
EB 3504

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**