

Loops

Lecture 3 of TDA 540 Object-Oriented Programming



Unsubscribed 4 p139
© photo75/istockphoto

Jesper Cockx

Fall 2018

Chalmers University of Technology — Gothenburg University

About the lab assignments

Still looking for a lab partner?

Go to the forum at <https://groups.google.com/forum/#!forum/tda540-2018>.

Note: handing in your solutions is obligatory, going to the lab sessions is not.

**Last week: repeat & some
missing details**

Last week's topics

- The `main` method
- Input and output
 - With `System.out` + `Scanner`
 - With `JOptionPane`
- Primitive datatypes (`byte`, `short`, `int`, `long`, `float`, `double`, `boolean`, `char`)
- The `String` class
- Variables, naming rules & conventions
- `if`-statements
- How to use the Java standard library

Variable scoping

Variables can only be used in the block where they are declared:

```
if ( condition ) {  
    String someString = ...  
} else {  
    String someString = ...  
}  
System.out.println(someString)  
// Error: someString is not in scope!
```

The `char` type in Java

A `char` is one character (letter/digit/symbol)

`'a'`, `'Q'`, `'5'`, `'#'`, ...

Each character has a **numerical value**:

```
int x = 'a'; // x == 97
int y = 'z'; // y == 122
int z = 'A'; // z == 65
int w = 'Z'; // w == 90
int p = '0'; // p == 48
int q = '9'; // q == 57
```

Detecting 'cancel' in JOptionPane

```
String input = JOptionPane.showInputDialog(
    "Please press cancel");

if ( input == null ) {
    JOptionPane.showMessageDialog(null,
        "Well done!");
} else {
    JOptionPane.showMessageDialog(null,
        "You had ONE job...");
}
```

Using JOptionPane and Scanner together

```
String inputString =
    JOptionPane.showInputDialog(
        "Please enter two numbers");
Scanner input = new Scanner(inputString);

int number1 = input.nextInt();
int number2 = input.nextInt();

JOptionPane.showMessageDialog(null ,
    "Sum: " + (number1 + number2) );
```


Calculating with floating-point numbers

Numbers of type `double` (and `float`) have a **limited size** and can often only store an **approximation** of the real number.

Overflow error result is bigger than

$$\text{Double}.\text{MAX_VALUE} = 1.8 \cdot 10^{308}$$

Underflow error result is smaller than

$$\text{Double}.\text{MIN_VALUE} = 4.9 \cdot 10^{-324}$$

Rounding error result cannot be represented exactly, e.g. $1.0/3 = 0.333\dots$

Live coding: properties of triangles

Assignment: Given the length of two sides of a triangle and the angle between them, calculate the third side.

$$c = \sqrt{a^2 + b^2 - 2ab \cos \beta}$$

Also determine whether the triangle is isosceles (has two equal sides) or equilateral (has three equal sides).

Kahoot!

Variables and if-statements.

While loops

Reminder: two kinds of instructions

- *Atomic* instructions (e.g. increase x by 1, wait 1 second, launch missile, ...)

- *Control* instructions:

Sequence First do x , then do y

Choice If x is true, then do y , else do z

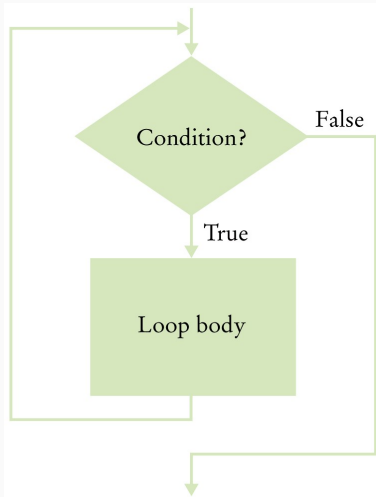
Iteration As long as x is true, repeat y

Jump Continue from point x

...

Structure of a while loop

```
while ( condition ) {  
    // code here is  
    // executed repeatedly  
    // until the condition  
    // becomes false  
}
```



Unnumbered 4 p157a
© John Wiley & Sons, Inc. All rights reserved.

Example: calculating the average

```
Scanner input = new Scanner(System.in);
final int MAX_COUNT = 10;
int count = 0;
double total = 0;
while ( count < MAX_COUNT ) {
    total += input.nextInt();
    count++;
}
System.out.println("Average: " + (total/count));
```

How to calculate average of *any* sequence?

15 min. break

Live coding: Heron's algorithm

Heron's algorithm is an iterative algorithm to calculate the square root of a number.

To calculate \sqrt{x} :

- Choose a desired precision $\epsilon > 0$
- Start with a first guess a
- As long as $|a^2 - x| > \epsilon$,
replace a by a new guess $(a + x/a)/2$

Example: calculating $\sqrt{25}$ with guess 12:

12.0, 7.042, 5.296, 5.008, 5.000007,...

Debugging: adding print statements

To find an error in your program, you can add `println` to print the intermediate results.

Kahoot!

While loops.

For loops

Structure of a for loop

```
for ( init ; condition ; increment ) {  
    // main body of the for loop  
}
```

Structure of a for loop

```
for ( init ; condition ; increment ) {  
    // main body of the for loop  
}
```

Equivalent while loop:

```
init;  
while ( condition ) {  
    // main body of the for loop  
    increment;  
}
```

Example: counting spaces in a string

```
String str = ...;
int spaces = 0;

for (int i = 0; i < str.length(); i++) {
    char current = str.charAt(i);
    if (current == ' ') {
        spaces++;
    }
}
```

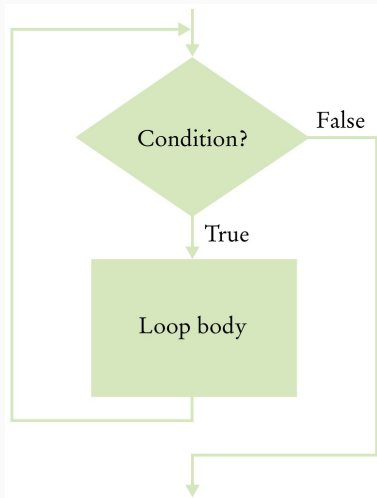
Other kinds of loops

Structure of a `do-while` loop

A `do-while` loop has the condition at the end:

```
do {  
    // main body of the  
    // do-while loop  
while ( condition );
```

The body will always be executed **at least once**.



Unnumbered 4 p157a
© John Wiley & Sons, Inc. All rights reserved.

Example: input validation without crashing

```
Scanner input = new Scanner(System.in);
int value;
do {
    System.out.print("Enter a number"
        + "between 1 and 100: ");
    value = input.nextInt();
} while (value < 1 || value > 100);
```

When to use which kind of loop

- Use a **for loop** if the number of iterations is known.
- Use a **do-while loop** if the loop body must be executed at least one.
- Use a **while loop** in all other cases.

Unstructured programming with `break`

You can end a loop early using `break`:

```
String str = ...;
for ( i = 0 ; i < str.length() ; i++ ) {
    if ( str.charAt(i) == 'x' ) {
        System.out.println("Found x at position " + i);
        break;
    }
}
```

Warning: using `break` can cause very unexpected errors and is considered bad form by most programmers!

Unstructured programming with `break`

Equivalent program without `break` (better!):

```
String str = ...;
int i = 0;
boolean found = false
while ( i < str.length() && !found ) {
    if ( str.charAt(i) == 'x' ) {
        found = true;
    }
    i++;
}
if ( found ) {
    System.out.println("Found x at position " + i);
}
```

Nested loops example: printing a chessboard pattern

```
for ( int row = 0 ; row < 8 ; row++ ) {  
    for ( int col = 0 ; col < 8 ; col++ ) {  
        if ( (row+col) % 2 == 0 ) {  
            System.out.print('#');  
        } else {  
            System.out.print('_');  
        }  
    } // end of row  
    System.out.print('\n');  
}
```

Nested loops example: printing a chessboard pattern

What happens if we change the `if` condition to...

- `row <= col`
- `(row+col) % 3 == 0`
- `(row*col) % 2 == 0`
- `(row+col) < 8`

How would you draw a diamond?

How about a disk?

Let your voice be heard!

Please go to

<http://bit.ly/tda540>

to give your opinion and help to improve this course.

What's next?

Next lecture: **methods and top-down design.**

Different lecturer: Krasimir Angelov

To do:

- Read the book:
 - Today: chapter 4
 - Next lecture: chapter 5
- Hand in the first lab (deadline Thursday)
- Start on the second lab:
programming a robot