Java Programming

— Basics of Java Programming: Loops (for statement) —

Waseda University

Problem

Example

Calculate the summation from 1 to 1000 and output the result on a screen.

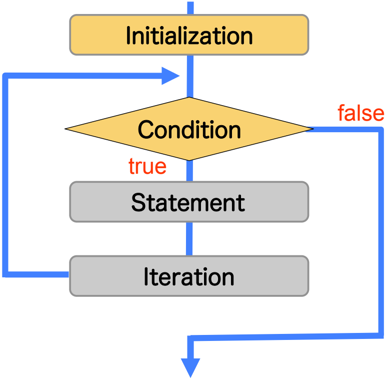
Sum1000.java

```java
public class Sum1000 {
  public static void main(String[] args) {
    int sum = 0;
    sum = sum + 1;
    sum = sum + 2;
    sum = sum + 3;
       ...
    sum = sum + 999;
    sum = sum + 1000;
    System.out.println("Summation from 1 to 1000: " + sum);
  }
}
```

Loop (`for` statement)

for statement

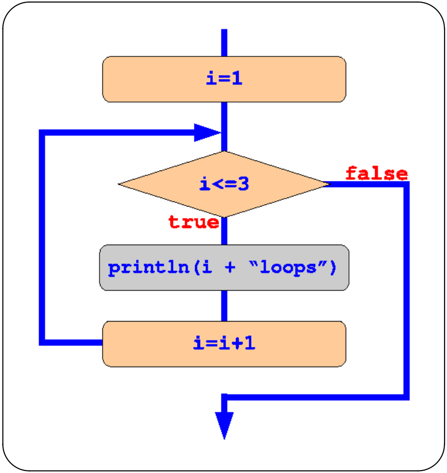
```
for (initialization; condition;  iteration) {
    statement
}
```



for statement (1)

```
int i;

for (i=1; i<=3; i=i+1) {
  System.out.println(i + "Times");
}
```



Results:

```
1Times
2Times
3Times
```

for statement (2)

```
int i;

for (i=1; i<=3; i=i+1)
  System.out.println(i + "Times");
```

- When the statement in { } is one, we can omit { }.

```
for (int i=1; i<=3; i=i+1) {         – Declare a variable i in for loop.
  System.out.println(i + "Times");– i is fine in for loop.
}
System.out.println(i);               – Error.
```

- You can declare variables inside for statement.
- In such cases, declared variables can be used inside for statement.

`for` statement (2)

```
int i;

for (i=1; i<=3; i=i+1)
  System.out.println(i + "Times");
```

- When the statement in `{ }` is one, we can omit `{ }`.

```
for (int i=1; i<=3; i=i+1) {        – Declare a variable i in for loop.
  System.out.println(i + "Times");– i is fine in for loop.
}
System.out.println(i);              – Error.
```

- You can declare variables inside `for` statement.
- In such cases, declared variables can be used inside `for` statement.

`for` statement (3)

In `example`....

```
int sum = 0;
sum = sum + 1;
sum = sum + 2;
   ...
sum = sum + 1000;
```

- You have to repeat to calculate “`sum = sum + i`” 1000 times while increasing the variable `i`.

⇓

```
int i, sum = 0;
for (i=1; i<=1000; i=i+1) {
  sum = sum + i;
}
```

Useful operator

Augmented assignment operator

`+=`, `-=`, `*=`, `/=`, `%=`

```
a += 10;
x *= 2;
```

$\Longleftrightarrow$

```
a = a + 10;
x = x * 2;
```

Increment and decrement operators

`++`, `--`

```
i++;
j--;
```

$\Longleftrightarrow$

```
i = i+1;
j = j-1;
```

In example....

```
for (i=1; i<=1000; i++) {
  sum += i;
}
```

$\Longleftrightarrow$

```
for (i=1; i<=1000; i=i+1) {
  sum = sum + i;
}
```

Useful operator

Augmented assignment operator

+=,   -=,   *=,   /=,   %=

```
a += 10;                      a = a + 10;
x *= 2;                ⇔      x = x * 2;
```

Increment and decrement operators

++,   --

```
i++;                          i = i+1;
j--;                   ⇔      j = j-1;
```

In example....

```
for (i=1; i<=1000; i++) {            for (i=1; i<=1000; i=i+1) {
  sum += i;                   ⇔        sum = sum + i;
}                                    }
```

Useful operator

Augmented assignment operator

```
+=,   -=,   *=,   /=,   %=
```

```
a += 10;
x *= 2;
```

$\Leftrightarrow$

```
a = a + 10;
x = x * 2;
```

Increment and decrement operators

```
++,   --
```

```
i++;
j--;
```

$\Leftrightarrow$

```
i = i+1;
j = j-1;
```

In example....

```
for (i=1; i<=1000; i++) {
  sum += i;
}
```

$\Leftrightarrow$

```
for (i=1; i<=1000; i=i+1) {
  sum = sum + i;
}
```

Example program (for statement)

SampleFor.java

```
public class SampleFor {
  public static void main (String[] args) {
    int i, n, sum;

    n = 1000;
    sum = 0;
    for (i=1; i<=n; i++) {
      sum += i;
    }
    System.out.println(" Sum from 1 to " + n + " is " + sum);
  }
}
```

[Results]

```
 Sum from 1 to 1000 is 500500
```

Nested loop

Triangle.java

```java
public class Triangle {
  public static void main(String[] args) {
    int i, j, n=5;
    for (i=1; i<=n; i++) {
      for (j=1; j<=i; j++) {
        System.out.print("*");
      }
      System.out.println();
    }
  }
}
```

[Results]

```
*
**
***
****
*****
```

Nested loop

```
int i, j, n=5;
for (i=1; i<=n; i++) {
  for (j=1; j<=i; j++) {
    System.out.print("*");
  }
  System.out.println();
}
```

The outer `for` statement

- It performs the statement inside `{ }` while changing the variable `i` from 1 to 5.
- `i` is `1`:

```
for (j=1; j<=1; j++) {
  System.out.print("*");
}
System.out.println();
```

Nested loop

```
int i, j, n=5;
for (i=1; i<=n; i++) {
   for (j=1; j<=i; j++) {
    System.out.print("*");
  }
  System.out.println();
}
```

The outer `for` statement

- It performs the statement inside `{ }` while changing the variable `i` from 1 to 5.
- `i` is 2:

```
for (j=1; j<=2; j++) {
  System.out.print("*");
}
System.out.println();
```

Nested loop

```
int i, j, n=5;
for (i=1; i<=n; i++) {
   for (j=1; j<=i; j++) {
    System.out.print("*");
  }
  System.out.println();
}
```

The outer `for` statement

- It performs the statement inside `{ }` while changing the variable `i` from 1 to 5.
- `i` is `5`:

```
for (j=1; j<=5; j++) {
  System.out.print("*");
}
System.out.println();
```