Selection

CONTROL STRUCTURES
SELECTION
(Slides adapted from Prof. H. Roumani)

PROF. Y. LESPÉRANCE
Dept. of Electrical Engineering & Computer Science

FLOW OF CONTROL
- What is control; how does it flow?
- Sequence versus Selection Flow

STATEMENT: IF

EXAMPLE
Write a method that reads an int and out-puts its abs value without using Math.abs.

```java
public int myAbs(int n) {
    int result = n;
    if (n < 0) {
        result = -n;
    }
    return result;
}
```
**PITFALL**

What is wrong with this?

```
public int myAbs(int n) {
    int result = n;
    if (n < 0) {
        result = -n;
    }
    return result;
}
```

**EXAMPLE**

Rewrite the `myAbs` body using if-else

```
int result;
if (n < 0) {
    result = -n;
} else {
    result = n;
}
return result;
```

**SELECTION: IF-ELSE**

```
if (condition) {
    Statement-A1
    Statement-A2
    ...
} else {
    Statement-B1
    Statement-B2
    ...
}
```

**PITFALL**

What is wrong with this?

```
if (n < 0) {
    int result = -n;
}
else {
    int result = n;
}
return result;
```
BUILDING THE CONDITION

- Relational Expression
  
  \[\text{if} \ (k < 0)\]

- Boolean Variable
  
  \[\text{boolean } b = k < 0 \]
  \[\text{if} \ (b)\]

- Boolean Expression
  
  \[\text{if} \ (k < 0 \ || \ b \ \&\& \ m > h)\]

Uses boolean operators: &&, ||, and !

OPERATOR PRECEDENCE

- . ++ -- !
- cast
- * / %
- + -
- < <= > >=
- == !=
- &&
- ||
- = op=

EXAMPLE

- Express the condition \(x \in [a, b]\)
  
  \[\text{if} \ (x \geq a \ \&\& \ x < b)\]

- Express the condition \(x \notin [a, b]\)
  
  \[\text{if} \ (! (x \geq a \ \&\& \ x < b))\]

Can use deMorgan’s Law to convert negated conjunctions to disjunctions.

\[\text{if} \ (x < a \ || \ x \geq b)\]

EXAMPLE

Given two object references \(x\) and \(y\), write a condition to determine if the objects they reference are equal.

Which one is correct and why:

- \[\text{if} \ (x == y)\]
- \[\text{if} \ (x.equals(y))\]
- \[\text{if} \ (x.equals(y) \ \&\& \ x != \text{null})\]
- \[\text{if} \ (x != \text{null} \ \&\& \ x.equals(y))\]
Implement these methods:

- **public boolean isOdd(int n)**
  *Returns true if n is odd else returns false.*

- **public char getFullGrade(int mark)**
  *Returns the full letter grade (A,B,C,D,E,F) given the mark (out of 100) in a course.*

- **public boolean isLeap(int year)**
  *Returns true if the year is a leap else returns false.*
  
  A leap year is divisible by 4 and if it is divisible by 100 then it must also be divisible by 400; e.g. 2016 is, 2000 is, but 2100 is not.
Iteration

**CONTROL STRUCTURES**

**ITERATION**

(Slides adapted from Prof. H. Roumani)

Prof. Y. Lespérance
Dept. of Electrical Engineering & Computer Science

---

**Iteration: for**

**Flow:**

S

for

{ initial
  condition
}

{ body }

bottom

condition

X

**Syntax:**

Statement-S
for (initial; condition; bottom)
{
  body;
}
Statement-X

**Algorithm:**

1. Start the for scope
2. Execute initial
3. If condition is false go to 9
4. Start the body scope
5. Execute the body
6. End the body scope
7. Execute bottom
8. If condition is true go to 4
9. End the for scope

---

**Example**

Output a table of square roots in [0,9].

```java
for (int i = 0; i < 10; i = i + 1) {
    double root = Math.pow(i, 0.5);
    System.out.print(i);
    System.out.print(" 	");
    System.out.println(root);
}
```

---

**Iteration Flow of Control**

(a)
## Iteration

**for (initial; condition; bottom)**

```java
for (int i = 0; i < MAX; i = i + 1)
{
    ...
}
```

**for (initial; condition; bottom)**

```java
int i = 0;
for (; i < MAX; i = i + 1)
{
    ...
}
```

### Can it be omitted?
- Can it be set to the literal `true`?
- What if it were false at the beginning?
- Is it monitored throughout the body?

## Control Structures/Lespérance

**for (initial; condition; bottom)**

- Can be any statement.
- Can be omitted.
- Will the loop become infinite if it is omitted?

### Example

Write a fragment to output the exponents of all powers of 2 that are smaller than a million.

**Correct output:**

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
```

```java
final int MILLION = 1000000;
for (int expo = 0; Math.pow(2, expo) < MILLION; expo++)
{
    System.out.print(expo);
    System.out.print(" ");
}
System.out.println();
```
EXAMPLE
Rewrite the fragment so it only outputs the exponent of the greatest power of 2 that is smaller than a million.

```java
int expo = 0;
for (; Math.pow(2, expo) < MILLION; expo++)
{
    System.out.println(expo - 1);
}
```

EXERCISES
Implement these methods:
- public boolean isPrime(int n)
  Returns true if n is prime, else returns false.
- public int log2(int n)
  Returns the number of repeated division of n by 2 until the result of the division is 1
- public double factorial(int n)
  Returns n!
- public double oddRecipSum(int n)
  Returns the sum of the reciprocals of odd integers with alternating signs between 1 and 10^6 (should around π/4)

NESTED LOOPS
- Must be fully nested
- Nested structures imply nested scopes

```java
for (int i = 0; i < 10; i++)
{
    for (int j = 0; j < 10; j++)
    {
        System.out.println(i + " " + j);
    }
}
```