Welcome to Software Foundations

CSE 5910

www.eecs.yorku.ca/course/5910

Instructor

- Name: Franck van Breugel
- Email: franck@cse.yorku.ca
  Please use your EECS or York account to send me email
- Office: Lassonde Building, room 3046
- Office hours: Mondays, 13:00-14:00, and Wednesdays and Fridays, 10:30-11:20 in the office, or by appointment

Evaluation

- Weekly: programming exercises (50%)
- March 21: programming test (10%)
- March 21: written test (10%)
- April 15: project (30%)

Programming

Practice, practice, practice, . . .

- weekly programming exercises
- programming exercises in the textbook

Drop deadline

March 7

Until this date you can drop the course without getting a grade for it and, hence, it will not affect your gpa.

www.registrar.yorku.ca/enrol/dates/fw13.htm contains important dates.

Academic honesty

"If you put your name on something, then it is your work, unless you explicitly say that it is not."

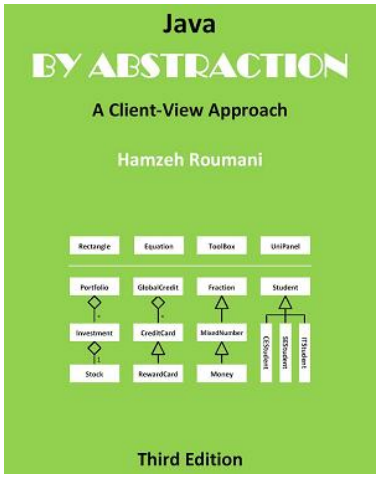
http://www.yorku.ca/grads/policies_procedures/academic_honesty.html contains more details.

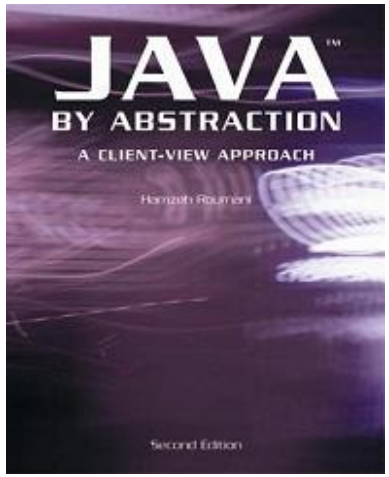
Textbook

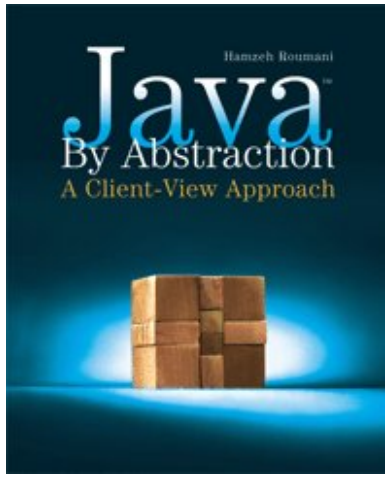
Hamzeh Roumani. *Java by Abstraction: A Client-View Approach*. First or second or third edition. Pearson, Toronto. 2005 and 2007 and 2010.

Why this textbook?

- As far as I know, this is the only textbook that uses the client-view (more about this view later in the course).
- The author is an award winning lecturer and teaches at York.







Textbook

Each chapter consists of

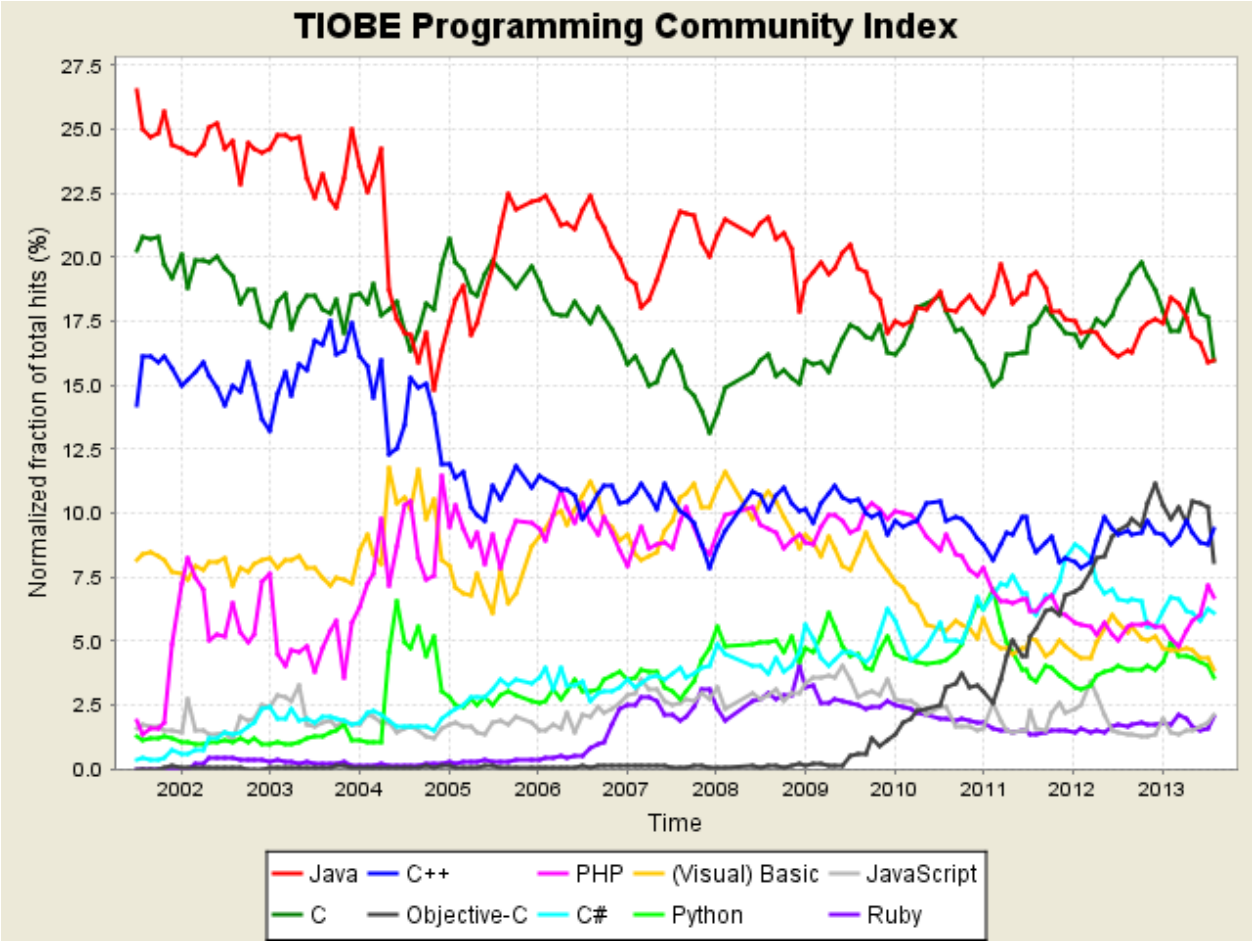
- reading material and contains several types of "boxes"
  - Programming Tip: essential
  - Java Details: useful
  - In More Depth: read and try to understand (don't worry if you don't understand them completely)
- review questions (try some yourself after having read a chapter),
- a lab (do it yourself),
- exercises (try some yourself, good preparation for tests), and
- eChecks.

Programming languages

ABC, Ada, Algol 60, Algol 68, Alice, APL, Basic, BPEL, C, C++, C#, Caml, COBOL, CSP, Eiffel, Emacs Lisp, Erlang, Esterel, Fortran, Haskell, IMP, Java, JavaScript, LaTeX, Lisp, LOTUS, Lustre, Maple, Mathematica, MATLAB, Mesa, Metafont, Miranda, ML, Modula-2, Oak, Oberon, Objective Caml, occam, Pascal, Perl, PHP, Pict, Pizza, PL/I, PostScript, Prolog, Promela, Python, Scala, Scheme, Simula, Smalltalk, SNOBOL, SOAP, Tcl, TeX, Turing, Visual Basic, Z, ...

Why Java?

Java is well-designed and popular.



source: `www.tiobe.com`

Why Java?

Because Java is popular, there are

- many textbooks about Java,
- many web pages about Java, and
- many software packages written in Java.

However, . . .

. . . this is *not* a course about Java.

Java is used to introduce you to programming and to explain several fundamental concepts.

Oak

In the early 1990s, James Gosling and some of his colleagues at Sun Microsystems, developed a programming language to program device controllers. The language was called Oak after an oak tree that stood outside Gosling's office. The language was expanded to a general purpose programming language and was renamed Java (because Oak was already a trademark). On May 23, 1995, Java was announced.

"Even though the Web had been around for 20 years or so, with FTP and telnet, it was difficult to use. Then Mosaic came out in 1993 as an easy-to-use front end to the Web, and that revolutionized people's perceptions. The Internet was being transformed into exactly the network that we had been trying to convince the cable companies they ought to be building. All the stuff we had wanted to do, in generalities, fit perfectly with the way applications were written, delivered, and used on the Internet. It was just an incredible accident. And it was patently obvious that the Internet and Java were a match made in heaven. So that's what we did."

James Gosling

James Gosling

James Gosling was born near Calgary in 1955. He received his BSc in computer science from the University of Calgary and his PhD from Carnegie Mellon University. In 2007, he was made an officer of the Order of Canada. He is best known as "the father of the Java programming language."

Trivial problem

Write a Java program that prints the Java's age.

Step 1

Solve the problem.

Step 2

Write the program.[1]

For this you need an editor:

- jEdit
- eclipse
- NetBeans
- Notepad
- ...

[1] Instead of program, we often call it an app(lication).

Name of a Java application

According to the Java Language Specification, the name of an application should be a sequence of letters and digits and the symbols _ and $, starting with a letter.

Code conventions

Rules how to write your code such as

- naming conventions for applications, variables, etc,
- indentation,
- etc.

Appendix C of the textbook contains the code conventions to which you and I will (try) adhere during this course. (We will not adhere to the rule for placement of braces.)

Another example of code conventions can be found at the URL babelfish.arc.nasa.gov/trac/jpf/wiki/devel/coding_conventions.

Code convention for application names

- Capitalize first letter.
- If the name is made up of more than one word, the capitalize the first letter of each.
- If the name is an acronym, then capitalize all letters.

Question

Which of the following class names adhere to the code convention?

- URL
- Sequence_Of_Characters
- Url
- CharacterSequence

Class header

```
public class AgeOfJava {
    public static void main(String[] args) {

    }
}
```

Class body

```
public class AgeOfJava {
    public static void main(String[] args) {

    }
}
```

Method header

```
public class AgeOfJava {
    public static void main(String[] args) {

    }
}
```

Method body

```
public class AgeOfJava {
    public static void main(String[] args) {
        ...
    }
}
```

Keywords

```
public class AgeOfJava {
    public static void main(String[] args) {

    }
}
```

Identifiers

```
public class AgeOfJava {
    public static void main(String[] args) {

    }
}
```

Separators

```
public class AgeOfJava {
    public static void main(String[] args) {

    }
}
```

Another code convention

Do not use magic numbers in expressions.

Magic numbers: numbers different from 0, 1, -1, 2, -2

Why should we not use magic numbers in expressions?

- Obscures the intent of the numbers
  ((100 * 60 * 60) / 9.58) / 1609.34
- Increases opportunities for subtle errors
- Makes it more difficult for the code to be adapted and extended in the future

Code convention for variables names

- Use lowercase letters.
- If the name is made up of more than one word, capitalize the first letter of each subsequent word.

Question

Which of the following variable names adhere to the code convention?

- myVariable
- MyVariable
- my_variable

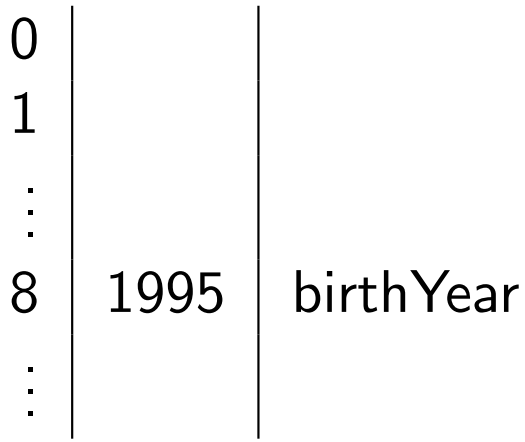
Memory model

int birthYear = 1995;

0		
1		
⋮		
8	00000000	birthYear
9	00000000	
10	00000111	
11	11001011	
⋮		

Memory model

$$\text{int birthYear} = 1995;$$



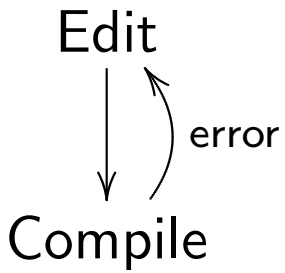
Terminology

System.out.println(age);

- System is (the name of) a class
- out is (the name of) a field
- println is (the name of) a method

Step 3

Compile the app: javac AgeOfJava.java

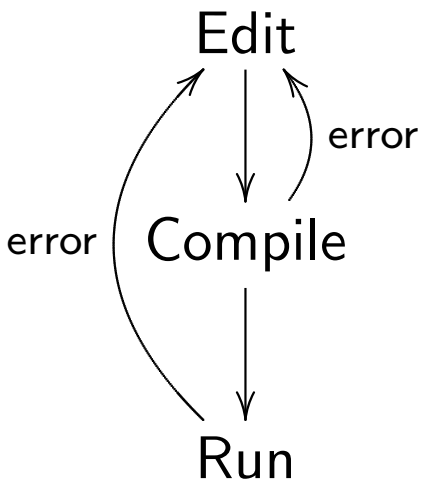


CSE 4302: Compilers and Interpreters

eclipse automatically compiles the app.

Step 4

Run the app: java AgeOfJava



In eclipse, you press the button with the green circle and the white arrow.

Nontrivial problem

By the end of this course you should be able to develop an app that

- randomly picks ten cities from a list of Ontarian cities,
- extracts the current temperature for each city from `weather.gc.ca`, and
- displays the temperatures.

The four steps

1. Solve the problem.
2. Write the app (using an editor such as eclipse).
3. Compile the app (done automatically in eclipse).
4. Run the app (pressing a button in eclipse).

Data types

How to represent and manipulate data?

Data types: definition

A data type consists of

- a name,
- a set of values, and
- a set of operations.

Data types: example

- name: int
- values: 0, 1, -1, 2, -2, ...
- operations: +, -, *, /, ...

How many values of type int are there?

Question: How many bytes are used to represent a value of type int ?

Answer:

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?

Answer: 4

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?

Answer: 4

Question: A byte consists of how many bits?

Answer:

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?

Answer: 4

Question: A byte consists of how many bits?

Answer: 8

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?
Answer: 4

Question: A byte consists of how many bits?
Answer: 8

Question: How many bits are used to represent a value of type int?
Answer:

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?

Answer: 4

Question: A byte consists of how many bits?

Answer: 8

Question: How many bits are used to represent a value of type int?

Answer: $4 \times 8 = 32$

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?
Answer: 4

Question: A byte consists of how many bits?
Answer: 8

Question: How many bits are used to represent a value of type int?
Answer: $4 \times 8 = 32$

Question: How many different values has a bit?
Answer:

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?
Answer: 4

Question: A byte consists of how many bits?
Answer: 8

Question: How many bits are used to represent a value of type int?
Answer: $4 \times 8 = 32$

Question: How many different values has a bit?
Answer: 2

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?
Answer: 4

Question: A byte consists of how many bits?
Answer: 8

Question: How many bits are used to represent a value of type int?
Answer: $4 \times 8 = 32$

Question: How many different values has a bit?
Answer: 2

Question: How many values of type int are there?
Answer:

How many values of type int are there?

Question: How many bytes are used to represent a value of type int?
Answer: 4

Question: A byte consists of how many bits?
Answer: 8

Question: How many bits are used to represent a value of type int?
Answer: $4 \times 8 = 32$

Question: How many different values has a bit?
Answer: 2

Question: How many values of type int are there?
Answer: 2^{32}

Data types: example

- name: int
- values: $[-2147483648, 2147483647]$
- operations: $+$, -, *, $/$, ...

Note that $2147483648 = 2^{31}$.

Operations

The operations are typed. For example,

$$\cdot - \cdot : (\mathrm{int} \times \mathrm{int}) \to \mathrm{int}$$

specifies that the operation $-$ takes two values of type int and returns a value of type int.

Overflow

What happens when you subtract one from -2147483648?

Data types: example

- name: int
- values: $[-2147483648, 2147483647]$
- operations:

$$\cdot + \cdot : (\text{int} \times \text{int}) \to \text{int}$$
$$\cdot - \cdot : (\text{int} \times \text{int}) \to \text{int}$$
$$\cdot * \cdot : (\text{int} \times \text{int}) \to \text{int}$$
$$\cdot / \cdot : (\text{int} \times \text{int}) \to \text{int}$$
$$\ldots$$

Slightly less trivial problem

Write an app that prints the age of Java as a real number.

Data types: example

- name: double
- values: 3.14, -7.3, ...
- operations:

$$\cdot + \cdot : (\text{double} \times \text{double}) \rightarrow \text{double}$$
$$\cdot - \cdot : (\text{double} \times \text{double}) \rightarrow \text{double}$$
$$\cdot * \cdot : (\text{double} \times \text{double}) \rightarrow \text{double}$$
$$\cdot / \cdot : (\text{double} \times \text{double}) \rightarrow \text{double}$$
$$\ldots$$

Casting

To convert an int to a double we use the operation

$$(\mathrm{double})\cdot : \mathrm{int} \to \mathrm{double}$$

This operation, known as casting, takes a value of type int and returns a corresponding value of type double.

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer:

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer: $8 \times 8 = 64$

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer: $8 \times 8 = 64$

Question: How many values of type double are there?
Answer:

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer: $8 \times 8 = 64$

Question: How many values of type double are there?
Answer: 2^{64}

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer: $8 \times 8 = 64$

Question: How many values of type double are there?
Answer: 2^{64}

Question: How many real number are there?
Answer:

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer: $8 \times 8 = 64$

Question: How many values of type double are there?
Answer: 2^{64}

Question: How many real number are there?
Answer: infinitely many

Double

A value of type double is represented by 8 bytes.

Question: How many bits is that?
Answer: $8 \times 8 = 64$

Question: How many values of type double are there?
Answer: 2^{64}

Question: How many real number are there?
Answer: infinitely many

Conclusion: most real numbers cannot be represented exactly

Casting

We distinguish between these two cases:

- $$(\text{double})\cdot : \text{int} \rightarrow \text{double}$$

  is an example of promotion. In general, promotions only lead to small round off errors or are precise.

- $$(\text{int})\cdot : \text{double} \rightarrow \text{int}$$

  is an example of demotion. In general, demotions loose information.

Casting

The compiler performs promotions automatically when needed.

Question

From the expression

$$(\text{double})\ \text{year} + (\text{double})\ (\text{currentDay} - \text{birthDay})\ /\ (\text{double})\ \text{daysPerYear}$$

which casts can be removed?

Another problem

Write an app that prints the age of Java as a real number with two digits precision.

Another problem

Write an app that prints the age of Java as a real number preceded by

```
The age of Java is
```

Data types: example

- name: String
- values: "zero or more characters"
- operations:

$$\cdot + \cdot : (\mathrm{String} \times \mathrm{String}) \rightarrow \mathrm{String}$$
$$\ldots$$

Another problem

Write an app that prints the age of Java as a real number preceded by

```
The "age" of Java is
```

Another problem

Write an app that prints the age of Java as a real number preceded by

`The age of Java is`

in Chinese.

Unicode

A Unicode is represented as

$$\backslash u????$$

where is each ? is one of the following:

$$0, 1, \ldots, 9, A, B, \ldots, F$$

For example, the Unicode for $\not<$ is \u226E.

Question: How many Unicodes are there?
Answer:

Unicode

A Unicode is represented as

$$\backslash u????$$

where is each ? is one of the following:

$$0, 1, \ldots, 9, A, B, \ldots, F$$

For example, the Unicode for $\not<$ is \u226E.

Question: How many Unicodes are there?
Answer: $16^4 = (2^4)^4 = 2^{16}$

Another problem

Write an app that prints the age of Java as a real number preceded by

```
The age of Java is
```

which not only gives the correct result today, but also tomorrow, the day after tomorrow, etc.

Importing packages

```
import franck.cse1020.Today;
```

- franck is a package
- cse1020 is a subpackage
- Today is a class

Another problem

Write an app that prints the age of Java as a real number preceded by

```
The age of Java is
```

which not only gives the correct result today, but also tomorrow, the day after tomorrow, etc, even if it is a leap year.

Remainder operator

The operator

$$\cdot \% \cdot : (\mathrm{int} \times \mathrm{int}) \to \mathrm{int}$$

yields the remainder of the division.

For example, the expression $2014 \ \% \ 4$ evaluates to 2.

Property

For all values a and b of type int,

$$(\mathrm{a} \ / \ \mathrm{b}) * \mathrm{b} + (\mathrm{a} \ \% \ \mathrm{b}) = \mathrm{a}$$

Data types: example

- name: boolean
- values: true, false
- operations:

$$\cdot \&\& \cdot : (\text{boolean} \times \text{boolean}) \to \text{boolean}$$
$$\cdot || \cdot : (\text{boolean} \times \text{boolean}) \to \text{boolean}$$
$$! \cdot : \text{boolean} \to \text{boolean}$$
$$\ldots$$

Some binary operations

$$
\begin{aligned}
&\cdot == \cdot : (\text{int} \times \text{int}) \to \text{boolean} \\
&\cdot < \cdot : (\text{int} \times \text{int}) \to \text{boolean} \\
&\cdot <= \cdot : (\text{int} \times \text{int}) \to \text{boolean} \\
&\ldots \\
&\cdot == \cdot : (\text{double} \times \text{double}) \to \text{boolean} \\
&\cdot < \cdot : (\text{double} \times \text{double}) \to \text{boolean} \\
&\cdot <= \cdot : (\text{double} \times \text{double}) \to \text{boolean} \\
&\ldots
\end{aligned}
$$

The expression $5 == 6$ evaluates to false and the expression $5 <= 6$ evaluates to true.

A ternary operation

The operation

$$\cdot ? \cdot : \cdot$$

of type

$$(\text{boolean} \times \text{int} \times \text{int}) \rightarrow \text{int}$$

is ternary, since it takes three arguments.

The expression $(5 == 6)\ ?\ 0 : 1$ evaluates to 1 and the expression $(5 <= 6)\ ?\ 0 : 1$ evaluates to 0.

Another problem

Write an app that prints the age of Java as a fraction preceded by

```
The age of Java is
```

which not only gives the correct result today, but also tomorrow, the day after tomorrow, etc, even if it is a leap year.

Sample question for a test

This question is based on material that has not been covered in the lectures, but can be found in the textbook.

Question

We can denote the steps that take place when evaluating the expression $(3 + 4) - 2$ as follows:

$$(3 + 4) - 2 \rightarrow 7 - 2 \rightarrow 5$$

Give the steps that take place when evaluating the expression

$$5 + (4 - 3)/5 - 2 * 3\%4$$

To do

- Study Chapter 1 of the textbook.
- Activate your EECS account: www.eecs.yorku.ca/activ8.
- Complete Check01C from the textbook by January 14.