Programming for Mobile Computing

EECS 1022

`moodle.yorku.ca`

Labs

For the things we have to learn before we can do them,
we learn by doing them.

Aristotle

During the labs, carefully read the instructions, try to follow them, consult the lecture slides, but also experiment and ask the teaching assistants for help. Work in teams of two as you can learn from each other.

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

What is the type of the attribute?

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

What is the type of the attribute?

Answer

`int`.

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

What is the type of the attribute?

Answer

`int`.

Question

What is a descriptive name for the attribute?

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

What is the type of the attribute?

Answer

`int`.

Question

What is a descriptive name for the attribute?

Answer

`counter`.

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

How do you declare an attribute?

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

How do you declare an attribute?

Answer

private *type name*;

Labs

Problem

An attribute that represents a counter that counts the number of times any of the four buttons is pressed. Use a descriptive name.

Question

How do you declare an attribute?

Answer

private *type name*;

Solution

private int counter;

Today's Lecture

- Classes
- Objects

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, $\ldots$
- operations: $+$, -, *, $/$, $\ldots$

How many values of type int are there?

Question

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

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?

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 bits are used to represent a value of type int?

How many values of type int are there?

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 bits are used to represent a value of type int?

Answer

$4 \times 8 = 32$

Question

How many different values has a bit?

How many values of type int are there?

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 values of type int are there?

How many values of type int are there?

Question

How many values of type int are there?

Answer

$$\underbrace{2 \times 2 \times \cdots \times 2}_{\text{32 times}} = 2^{32}$$

Data types: example

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

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.

Data types: example

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

$$\begin{aligned}
\cdot + \cdot &: (\mathrm{int} \times \mathrm{int}) \to \mathrm{int} \\
\cdot - \cdot &: (\mathrm{int} \times \mathrm{int}) \to \mathrm{int} \\
\cdot * \cdot &: (\mathrm{int} \times \mathrm{int}) \to \mathrm{int} \\
\cdot / \cdot &: (\mathrm{int} \times \mathrm{int}) \to \mathrm{int} \\
\ldots
\end{aligned}$$

Data types: example

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

$$\begin{aligned}
&\cdot\&\&\cdot : (\text{boolean} \times \text{boolean}) \to \text{boolean}\\
&\cdot||\cdot : (\text{boolean} \times \text{boolean}) \to \text{boolean}\\
&!\cdot : \text{boolean} \to \text{boolean}
\end{aligned}$$

Primitive types

Question

What are the names of the five most used primitive types?

Primitive types

Question

What are the names of the five most used primitive types?

Answer

boolean, char, double, int and long.[a]

[a] The other three, less used, primitive types are byte, float and short.

Location of the robot

Question

Can the location of the robot be represented as a single value of primitive type?

Location of the robot

Question

Can the location of the robot be represented as a single value of primitive type?

Answer

No.

Programming Paradigms

- Object-oriented programming
- Imperative programming
- Functional programming
- Logic programming
- Concurrent programming
- Event-driven programming
- Constraint programming
- . . .

Object-Oriented Programming

Objects as a formal concept in programming were introduced in the 1960s in programming language Simula 67. This language was created by Ole-Johan Dahl and Kristen Nygaard of the Norwegian Computing Center in Oslo.

Ole-Johan Dahl

Ole-Johan Dahl (October 12, 1931 – June 29, 2002) was a Norwegian computer scientist and is considered to be one of the fathers of object-oriented programming.

source: ifi.uio.no

Kristen Nygaard

Kristen Nygaard (August 27, 1926 – August 10, 2002) was a Norwegian computer scientist and is considered to be one of the fathers of object-oriented programming.

source: ifi.uio.no

Dahl and Nygaard

In 2001, Ole-Johan Dahl and Kristen Nygaard won the Turing award.

The A.M. Turing Award is given annually by the Association for Computing Machinery (ACM) to "an individual selected for contributions of a technical nature made to the computing community." The Turing Award is recognized as the "highest distinction in Computer Science" and "Nobel Prize of computing."

source: ifi.uio.no

Advantages of OOP

- easy to re-use code
- easy to extend code
- easy to maintain code
- easy to test code
- fits well with the real world
- ...

However, (some of) these advantages are debatable.

Mordechai Ben-Ari. Objects never?: well, hardly ever! *Communications of the ACM*, 53(9): 32–35, September 2010.

Location of the robot

Question

How can we represent the location of the robot? (You may use multiple values of primitive type.)

Location of the robot

Question

How can we represent the location of the robot? (You may use multiple values of primitive type.)

Answer

Its x- and y-coordinate.

Location of the robot

Question

How can we represent the location of the robot? (You may use multiple values of primitive type.)

Answer

Its x- and y-coordinate.

Question

For each datum, what is a descriptive name and an appropriate type?

Location of the robot

Question

How can we represent the location of the robot? (You may use multiple values of primitive type.)

Answer

Its x- and y-coordinate.

Question

For each datum, what is a descriptive name and an appropriate type?

Answer

- x : int
- y : int

How to represent locations?

Question

How to represent the location (1, 2)?

How to represent locations?

Question

How to represent the location (1, 2)?

Answer

$$\begin{array}{c|c|} \cline{2-2} x & 1 \\ y & 2 \\ \cline{2-2} \end{array}$$

How to represent locations?

Question

How to represent the location (1, 2)?

Answer

$$\begin{array}{c|c|}\cline{2-2} x & 1 \\ y & 2 \\ \cline{2-2}\end{array}$$

Question

How to represent the location (3, 4)?

How to represent locations?

Question

How to represent the location (1, 2)?

Answer

$$\begin{array}{c|c|} x & 1 \\ y & 2 \end{array}$$

Question

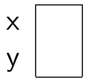
How to represent the location (3, 4)?

Answer

$$\begin{array}{c|c|} x & 3 \\ y & 4 \end{array}$$

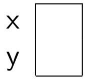
How to represent locations?

All locations are an instance of the following pattern.



How to manipulate locations?

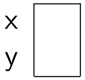
If you are given an instance of the pattern



what kind of questions may you want to ask about this data?

How to manipulate locations?

If you are given an instance of the pattern

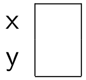


what kind of questions may you want to ask about this data?

- What is the x-coordinate of this location?

How to manipulate locations?

If you are given an instance of the pattern

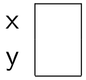


what kind of questions may you want to ask about this data?

- What is the x-coordinate of this location?
- What is the y-coordinate of this location?

How to manipulate locations?

If you are given an instance of the pattern

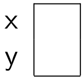


what kind of questions may you want to ask about this data?

- What is the x-coordinate of this location?
- What is the y-coordinate of this location?
- How does the location change if the robot moves North?

How to manipulate locations?

If you are given an instance of the pattern



what kind of questions may you want to ask about this data?

- What is the x-coordinate of this location?
- What is the y-coordinate of this location?
- How does the location change if the robot moves North?
- How does the location change if the robot moves South?
- ...

Objects and classes

Question

What is an object?

Answer

"An instance of a class."

Question

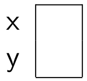
What is a class?

Answer

"A blueprint for objects."

You often find these circular definitions in textbooks and on the Internet, but they are not particularly helpful.

What is a class?



A class contains attributes. Each attribute has a name and a type.

x : int
y : int

What is a class?

- What is the x-coordinate of this location?
- What is the y-coordinate of this location?
- . . .

A class contains methods. Each method has a signature and possibly a return type.

getX() : int
getY() : int

What is an object?

An object is an instance of a class.

An object has a state. The state of an object consists of the attributes of the class and their values.

x	1
y	2

What is an object?

An object has an identity. This identity is unique. That is, two different objects have different identities.

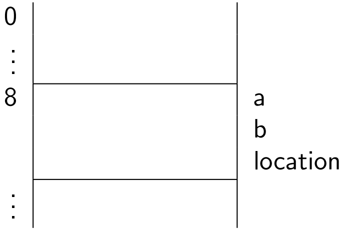
This is an abstract notion. In more concrete terms, you may think of an object's identity as the address in memory where it is stored. Obviously, two different objects cannot be stored at the same memory address.

How to create objects?

```
int  a = 1;
int  b = 2;
Location location = new Location(a, b);
```

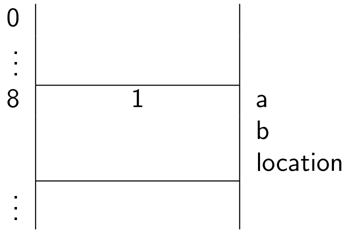
How to create objects?

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```



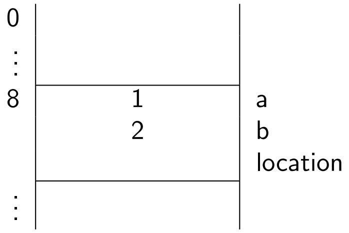
How to create objects?

```
int a = 1;
 int  b = 2;
Location  location  = new Location(a, b);
```



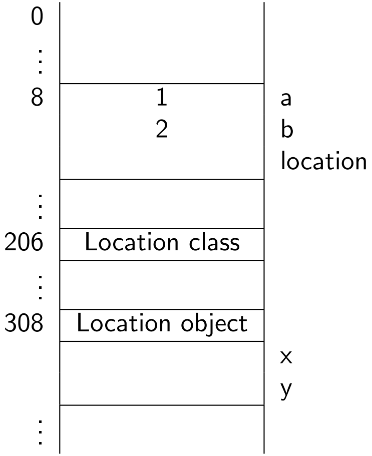
How to create objects?

```
int a = 1;
int b = 2;
Location  location  = new Location(a, b);
```



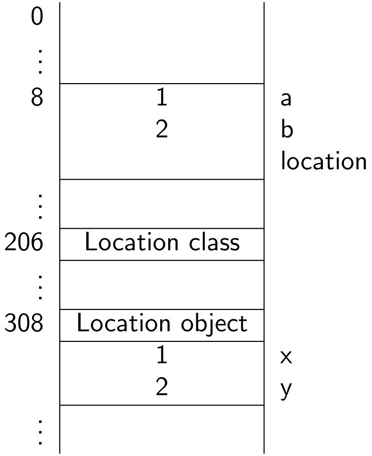
How to create objects?

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```



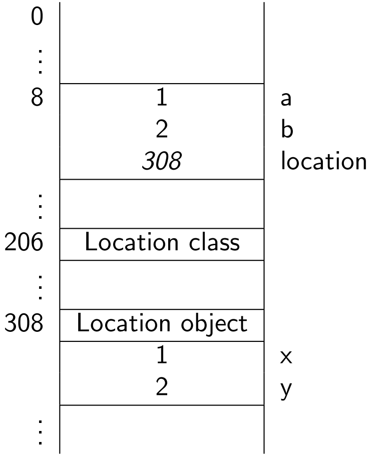
How to create objects?

```
int a = 1;
int b = 2;
Location  location  = new Location(a, b);
```



How to create objects?

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```



Object creation in memory model

- The first time we encounter a class, we allocate a block in memory for the class.
- Whenever we encounter new, we allocate a block in memory for the object.
- Whenever we encounter a constructor, we initialize the attributes by putting the values of the attributes in the block of the object.

Terminology

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```

- location is the name of a

Terminology

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```

- location is the name of a variable.

Terminology

```
int  a = 1;
int  b = 2;
Location location  = new Location(a, b);
```

- location is the name of a variable.
- the type of the variable location is

Terminology

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```

- location is the name of a variable.
- the type of the variable location is Location.

Terminology

```
int a = 1;
int b = 2;
Location location = new Location(a, b);
```

- location is the name of a variable.
- the type of the variable location is Location.
- location is also called an object reference.

Types

We distinguish between

- primitive types: boolean, char, double, int, long, (byte, float, short) and
- reference types: classes

Invoking a method

Consider the method
`public type methodName(type`$_1$ `parameterName`$_1$`, ..., type`$_n$ `parameterName`$_n$`)`
in the class `ClassName`.

This method is invoked as
`objectReference.methodName(argument`$_1$`, ..., argument`$_n$`)`
where the type of objectReference is `ClassName` and `argument`$_i$ is (compatible with) `type`$_i$.

Location API

The API of the Location class can be found at the URL
http://www.eecs.yorku.ca/course_archive/2016-17/S/1022/api/location.api/franck/Location.html

Invoking a method

Question

How do you create a $\mathrm{Location}$ object named first located at (1,2) and get the location north of it and name that object second?

Invoking a method

Question

How do you create a Location object named first located at (1,2) and get the location north of it and name that object second?

Answer

```
Location first  = new Location(1, 2);
Location second = first.north();
```

Invoking a method

Question

Draw the diagram representing the memory once the execution has reached the end of the following snippet.

```
Location first  = new Location(1, 2);
Location second = first.north();
```

Answer

⋮		
8	*308*	first
	408	second
⋮		
206	Location class	
⋮		
308	Location object	
	1	x
	2	y
⋮		
408	Location object	
	1	x
	1	y
⋮		

Printing a location

Question

Is there a method we can use to print the result?

Printing a location

Question

Is there a method we can use to print the result?

Answer

Yes, `public String toString()`

Printing a location

Question

Is there a method we can use to print the result?

Answer

Yes, `public String toString()`

Question

How do we invoke this method?

Printing a location

Question

Is there a method we can use to print the result?

Answer

Yes, `public String toString()`

Question

How do we invoke this method?

Answer

`String result = location.toString()`

Printing a location

```
Location first = new Location(1, 2);
Location second = first.north();
String result = second.toString();
output.println(result);
```

Printing a location

Exercise

Draw the diagram representing the memory once the execution has reached the end of the snippet on the previous slide.

Solution to exercise

⋮		
8	*308*	first
	408	second
	682	result
⋮		
206	Location class	
⋮		
308	Location object	
	1	x
	2	y
⋮		
408	Location object	
	1	x
	1	y
⋮		
534	String class	
682	String object	
	"(1,1)"	

. . . objects versus object references

Question

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

At the end of the execution of the above snippet, how many objects are there and how many objects references are there?

. . . objects versus object references

Question

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

At the end of the execution of the above snippet, how many objects are there and how many objects references are there?

Answer

Four objects and six object references.

...objects versus object references

Question

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

At the end of the execution of the above snippet, how many objects are there and how many objects references are there?

Answer

Four objects and six object references.

Exercise

Draw the diagram representing the memory once the execution has reached the end of the above snippet.

Solution to exercise

100	*300*	f
	400	g
	500	h
	600	i
	400	j
	400	k
200	Location class	
300	Location object	
	0	x
	0	y
400	Location object	
	0	x
	0	y
500	Location object	
	1	x
	2	y
600	Location object	
	0	x
	2	y

When are two objects references the same?

What do we mean by the same?

- Do they refer to the same object, that is, do they have the same identity?
- Do they refer to objects with the same state, that is, do their attributes have the same values?

When are two objects references the same?

What do we mean by the same?

- Do they refer to the same object, that is, do they have the same identity?
- Do they refer to objects with the same state, that is, do their attributes have the same values?

```
Location one = ...
Location other = new Location(1, 1);
boolean identical = (one == other);
boolean same = one.equals(other);
```

When are two objects references the same?

Question

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

Fill the following table with true (T) and false (F).

==	f	g	h	i	j	k
f						
g						
h						
i						
j						
k						

When are two objects references the same?

Answer

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

==	f	g	h	i	j	k
f	T	F	F	F	F	F
g	F	T	F	F	T	T
h	F	F	T	F	F	F
i	F	F	F	T	F	F
j	F	T	F	F	T	T
k	F	T	F	F	T	T

When are two objects references the same?

Question

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

Fill the following table with true (T) and false (F).

equals	f	g	h	i	j	k
f						
g						
h						
i						
j						
k						

When are two objects references the same?

Answer

```
Location f = new Location();
Location g = new Location();
Location h = new Location(1, 2);
Location i = new Location(0, 2);
Location j = g;
Location k = j;
```

equals	f	g	h	i	j	k
f	T	T	F	F	T	T
g	T	T	F	F	T	T
h	F	F	T	F	F	F
i	F	F	F	T	F	F
j	T	T	F	F	T	T
k	T	T	F	F	T	T