Classes in C++

- Version 1: Dr. Ofir Pele
- Version 2: Dr. Miri Ben-Nissan
- Version 3: Dr. Erel Segal-Halevi

	C	C++	Jawa
Keyword	struct	class or struct	class
Filename	any (usually: name.h)	any (usually: name.hpp name.cpp)	name.java
Attributes	Yes	Yes	Yes
Methods	No	Yes	Yes
Access control	all public	public or private	public or private
Memory	stack	stack	heap
Operators	No	Yes	No

structs and classes

Where did structs go?

 In C++ class==struct, except that by default struct members are public and class members are private:

```
int main()
struct MyStruct
                          MyStruct s;
   int x;
                          s.x = 1; // ok
                          MyClass c;
class MyClass
                          c.x = 1; // error
   int x;
```

structs & classes (folder 1)

```
All of these are the same:
                                All of these are the same (and useless):
                                class A
struct A
   int x;
                                    int x;
};
                                };
struct A
                                class A
   public:
                                    private:
   int x;
                                    int x;
};
                                };
class A
                                struct A
                                    private:
   public:
   int x;
                                    int x;
                                };
};
```

Arrangement of Classes in Memory

- Version 1: Dr. Ofir Pele
- Version 2: Dr. Miri Ben-Nissan
- Version 3: Dr. Erel Segal-Halevi

double im) {...}

};

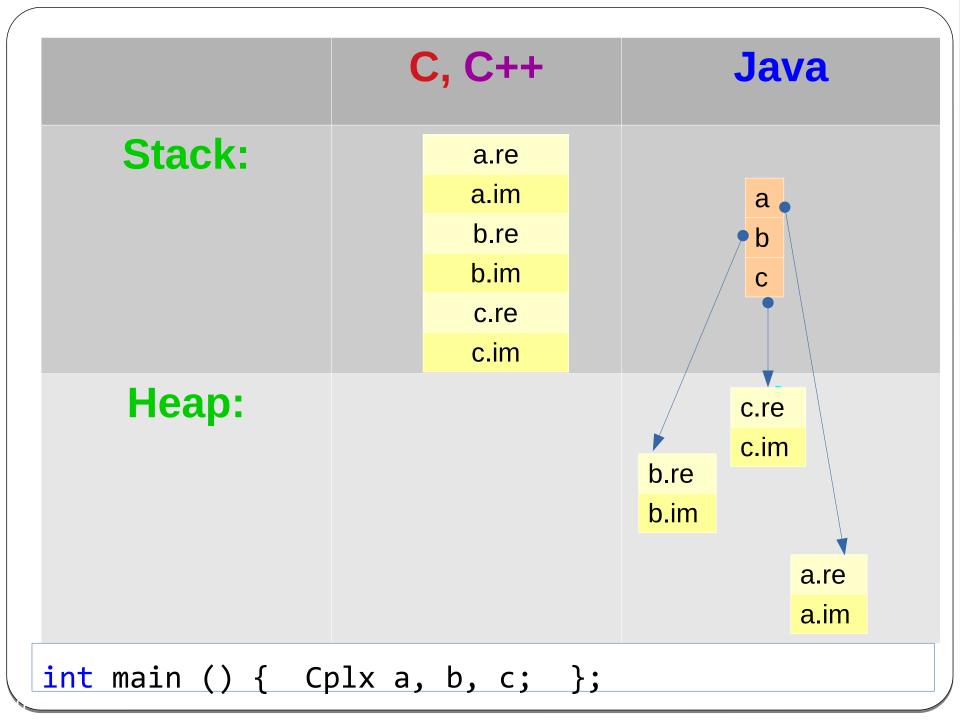
double im) {...}

};

};

{...}

```
Java
                 int main() {
                                    void main(...) {
int main() {
                    Cplx a(5,10);
                                       Cplx a =
   Cplx a;
                                       new
   a.re=5;
                                         Cplx(5,10);
   a.im=10;
```



Two ways to implement a method (folder 2)

```
class Complex {
  double re, im;
public:
Complex () { re=0; im=0; }// inline constructor
Complex (double re, double im); // "outline"
Complex sum (Complex b) { return
Complex(a.re+b.re, a.im+b.im); } // inline method
Complex diff (Complex b);  // "outline"
};
```

Implementing methods out-of-line

```
Complex::Complex (double re, double im) {
    this > re = re;
    this > im = im;
}
```

The address of the instance for which the member method was invoked.

```
Complex Complex::diff(Complex b) {
    return Complex(a.re-b.re, a.im-b.im);
}
```

```
Class Basics – member/static (folder 3)
class List
public:
   static int getMaxSize();
   int getSize();
   // static int max size=1000; //error! (declare outside)
   int size=0;
};
int List::max_size=1000; //ok, in one cpp file
int main()
   List 1;
   1.getSize();
   List::getMaxSize();
   l.getMaxSize(); //compiles ok, but bad style
```

this

```
static int List::getMaxSize() //no this!
{
    return this->size; // compile error!
    return max_size; // ok
}
int List::getSize()
{
    return this->size; //ok
}
```

What file-names should we use?

- The C++ compiler does not care how your files are called.
- It is common to put a class declaration in file ClassName.hpp (or ClassName.h) and the class implementation in file ClassName.cpp.
- Why is it better?
 - Hiding implementation details.
 - Saving comiplation time when you have a good **Makefile** (see folder 4).