



# <page-header><text><section-header><text><text><text>



**UCI** 

6

### **UCL Programming Strategy** An approach to problem solving · Prior to OOP, languages only had procedures/ · Decompose a problem into a sequence of simpler functions. methods and then call each method. • For each simpler method repeat the process. · This led to a style of programming known by various names: · Stop when a method becomes trivial to write. - Procedural decomposition - E.g., doing payroll processing. - Top-down programming - Structured programming - and others © 2005, Graham Roberts 5



**UCL** 

a

### DEPARTMENT OF COMPUTER SCIENCE

## Method size

- Keep methods small:
   "Small enough but no smaller."
- If a method is longer than ~15 lines then always consider splitting it.
- Don't go mad and reduce all methods to one line though!
- Develop your feel for what makes a good method.

© 2005, Graham Roberts

## <text><section-header><list-item><list-item><list-item><list-item><list-item><list-item><list-item>

DEPARTMENT OF COMPUTER SCIENCE	⁺UCL
Questions?	
© 2005. Graham Roberts	

A Simple One Class Example	
<ul> <li>Program specification: <ul> <li>Read in a sequence of integers, sort them, and output the sorted sequence.</li> </ul> </li> <li>The problem can be simply decomposed into three methods: read, sort and output.</li> </ul>	
© 2005, Graham Roberts	12









DEPARTMENT OF COMPUTER SCIENCE	UCL
sortIntegers	
<ul> <li>Now needs to take array parameter and return a array:</li> </ul>	n
public int[] sortIntegers(int[] myInts)	
{ // Do the sorting return mvInts:	
}	
© 2005, Graham Roberts	17

DEPARTMENT OF COMPUTER SCIENCE	
outputIntegers	
<ul> <li>Only needs to have an array parameter: public void outputIntegers(int[] myInts) { // loop to output integers     }     }     </li> </ul>	
© 2005, Graham Roberts	18





DEPARTMENT OF COMPUTER SCIENCE	L
Probably	
<ul> <li>readInteger now only does <i>one</i> thing (read in 100 integers).</li> <li>Having to create the array made it a little less general purpose, less cohesive.</li> <li>Of course, the array size is fixed, so the next variant should change that (an exercise for you).</li> </ul>	
© 2006, Gasham Roberts	21













DEPARTMENT OF COMPUTER SCIENCE	⁴UCL
Responsibilities	
<ul> <li>The <i>object</i> is responsible for holding the arra</li> <li>The object provides the services of reading, and displaying data to other objects.</li> </ul>	ay. sorting
© 2005, Graham Roberts	20

DEPARTMENT OF COMPUTER SCIENCE	⁺UCL
Questions?	
© 2005, Graham Roberts	30



**UCL** 

**Specifying a Contract** 

a comment.

syntax.

© 2005, Graham Roberts

• The order of method call can be specified using

- Or other documentation, such as Javadoc.

• Design and programming issue.

- Programmer must get it right!

· Unfortunately, can't be enforced via language



But messy and needs extra code.

}

}

DEPARTMENT OF COMPUTER SCIENCE	⁺UCL	DEPARTMENT OF COMPUTER SCIENCE	≜UC
Rethink interface		Public/Private	
<ul> <li>Why are read, sort and display meth public?</li> <li>Make them private!</li> <li>Provide a single public method to us</li> </ul>	nods se object	<pre>private void readIntegers() { } // private void sortIntegers() { } // private void outputIntegers() { } // public void readSortOutput() // new {     readIntegers(); // Enforce co     sortIntegers();     outputIntegers(); }</pre>	<sup>1</sup> Can only be used by <sup>1</sup> other methods in <sup>1</sup> same class. <sup>1</sup> control method rrect order.
© 2005, Graham Roberts	25	© 2005, Graham Roberts	

33

**UCI** 

36

34

**UCI** 

32





DEPARTMENT OF COMPUTER SCIENCE	JCL
Parameters v. instance variables?	
<ul> <li>In this 1-class example parameters were eliminat <ul> <li>Nature of the problem.</li> </ul> </li> <li>Parameters still essential: <ul> <li>Function-like methods.</li> <li>Generic methods.</li> <li>Passing information from one object to another.</li> </ul> </li> </ul>	ed.
© 2005, Graham Roberts	41



**UCL** 

40





DEPARTMENT OF COMPUTER SCIENCE	≜UCL	DEPARTMENT OF COMPUTER SCIENCE	≜UC
No		A better way	
<ul> <li>Top-down decomposition is a poor way for solving larger problems and for designing larger programs.</li> <li>In fact, it's often a Complete disaster!</li> <li>The approach simply does not scale-up.</li> </ul>		<ul> <li>Object-oriented design and provides much better solution construction of larger program</li> <li>As the year continues we will detail.</li> <li>But also read the text book not</li> </ul>	rogramming s to the is. examine why in ow.
© 2005, Graham Roberts	45	© 2005, Graham Roberts	

EPARTMENT OF COMPUTER SCIENCE	≜UCL	DEPARTMENT OF COMPUTER SCIENCE
For now		Summary
<ul> <li>Let's consider route planning</li> <li>Top-down solution is rigid, inflexible ar</li> <li>OO solution is general purpose, flexib</li> </ul>	nd fragile. le and robust.	<ul> <li>Programs can methods that i</li> <li>Methods shou</li> <li>Method based solving small s only.</li> <li>But methods a classes in object</li> </ul>
0 2005, Graham Roberts	47	© 2005, Graham Roberts

© 2005

F COMPUTER SCIENCE	<b>≜UCL</b>
mary	
ograms can be decomposed into a collection ethods that call one another.	on of
ethods should be small and cohesive. ethod based decomposition is suitable for living small scale problems or sub-problems ily.	6
ut methods are an essential building block o asses in object-oriented programs.	of