

Introduction to  
Object-Oriented Programming  
Using C++

Peter Müller

*pmueller@uu-gna.mit.edu*

Globewide Network Academy (GNA)  
[www.gnacademy.org/](http://www.gnacademy.org/)

November 18, 1996



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>A Survey of Programming Techniques</b>	<b>3</b>
2.1	Unstructured Programming . . . . .	3
2.2	Procedural Programming . . . . .	4
2.3	Modular Programming . . . . .	5
2.4	An Example with Data Structures . . . . .	6
2.4.1	Handling Single Lists . . . . .	6
2.4.2	Handling Multiple Lists . . . . .	8
2.5	Modular Programming Problems . . . . .	8
2.5.1	Explicit Creation and Destruction . . . . .	9
2.5.2	Decoupled Data and Operations . . . . .	9
2.5.3	Missing Type Safety . . . . .	10
2.5.4	Strategies and Representation . . . . .	10
2.6	Object-Oriented Programming . . . . .	11
2.7	Exercices . . . . .	12
<b>3</b>	<b>Abstract Data Types</b>	<b>13</b>
3.1	Handling Problems . . . . .	13
3.2	Properties of Abstract Data Types . . . . .	15
3.3	Generic Abstract Data Types . . . . .	17
3.4	Notation . . . . .	17
3.5	Abstract Data Types and Object-Orientation . . . . .	18
3.6	Exercices . . . . .	19
<b>4</b>	<b>Object-Oriented Concepts</b>	<b>21</b>
4.1	Implementation of Abstract Data Types . . . . .	21
4.2	Class . . . . .	23
4.3	Object . . . . .	24
4.4	Message . . . . .	24
4.5	Summary . . . . .	25
4.6	Exercices . . . . .	26

<b>5</b>	<b>More Object-Oriented Concepts</b>	<b>27</b>
5.1	Relationships	27
5.2	Inheritance	30
5.3	Multiple Inheritance	32
5.4	Abstract Classes	34
5.5	Excercises	36
<b>6</b>	<b>Even More Object-Oriented Concepts</b>	<b>39</b>
6.1	Generic Types	39
6.2	Static and Dynamic Binding	41
6.3	Polymorphism	42
<b>7</b>	<b>Introduction to C++</b>	<b>47</b>
7.1	The C Programming Language	47
7.1.1	Data Types	47
7.1.2	Statements	49
7.1.3	Expressions and Operators	50
7.1.4	Functions	54
7.1.5	Pointers and Arrays	55
7.1.6	A First Program	56
7.2	What Next?	57
<b>8</b>	<b>From C To C++</b>	<b>59</b>
8.1	Basic Extensions	59
8.1.1	Data Types	59
8.1.2	Functions	62
8.2	First Object-oriented Extensions	63
8.2.1	Classes and Objects	63
8.2.2	Constructors	65
8.2.3	Destructors	68
<b>9</b>	<b>More on C++</b>	<b>69</b>
9.1	Inheritance	69
9.1.1	Types of Inheritance	70
9.1.2	Construction	70
9.1.3	Destruction	72
9.1.4	Multiple Inheritance	72
9.2	Polymorphism	72
9.3	Abstract Classes	74
9.4	Operator Overloading	74
9.5	Friends	76
9.6	How to Write a Program	77
9.6.1	Compilation Steps	78
9.6.2	A Note about Style	79
9.7	Excercises	79

<b>10 The List – A Case Study</b>	<b>81</b>
10.1 Generic Types (Templates)	81
10.2 Shape and Traversal	83
10.3 Properties of Singly Linked Lists	83
10.4 Shape Implementation	85
10.4.1 Node Templates	85
10.4.2 List Templates	87
10.5 Iterator Implementation	90
10.6 Example Usage	93
10.7 Discussion	93
10.7.1 Separation of Shape and Access Strategies	93
10.7.2 Iterators	94
10.8 Exercises	95
<b>Bibliography</b>	<b>97</b>
<b>A Solutions to the Exercises</b>	<b>99</b>
A.1 A Survey of Programming Techniques	99
A.2 Abstract Data Types	100
A.3 Object-Oriented Concepts	102
A.4 More Object-Oriented Concepts	103
A.5 More on C++	105
A.6 The List – A Case Study	105

[Click here to download full PDF material](#)