

Object Oriented Programming with Smalltalk

by Bryce Hendrix

Disclaimer: This document is provided for educational purposes, and contains some non-original content. If you are the author of any material included in this document and would like it removed, please send mail to Bryce Hendrix at hendrix@engr.arizona.edu ***and it will be removed immediately. Contents of this document cannot be used for commercial usage without explicit permission from the author.***

<i>Lecture 1 : What is an Object?</i>	7
• 2 Rules of Smalltalk	7
• What's special about an Object?	7
• OO versus Procedural Approach to programming	7
<i>Lecture 2: Classes and Instances</i>	11
• Class	11
• Instance	11
• Class Hierarchy	11
<i>Lecture 3: Messages, Methods, and Programming in Smalltalk</i>	13
• Messages	13
• Methods	13
• Programming in Smalltalk	14
<i>Lecture 4: OO Classification Techniques</i>	15
• Specialization	15
• Abstraction	16
• Composition	17
• Factorization	18
<i>Lecture 5: Encapsulation & Polymorphism</i>	20
• Encapsulation	20
• Polymorphism	21
<i>Lecture 6: OO 4-Pass Process – an Investment Manager</i>	22
• Pass 1: Abstraction	22
• Pass 2: Abstraction	22
• Pass 3: Composition	22
• Pass 4: Factorization	22
<i>Lecture 7: The Object Class</i>	27
• Functionality of an object	27
• Comparison of objects	27
• Copying objects	27
• Accessing indexed variables	28
<i>Lecture 8: Messages & Methods</i>	30
• Message Expressions	30
• Method Lookup	30
<i>Lecture 9: Variables and Return values</i>	32

• Method arguments	32
• Temp variables	32
• Instance variables	32
• Class instance variables	32
• Class Variables	33
• Global Variables	33
• Return Values	33
<i>Lecture 10: Blocks and Branching</i>	34
• Blocks	34
• Class Boolean	34
• Branching (Control Structures)	34
<i>Lecture 11: Reporting Errors and Debugging techniques</i>	36
• Error Handling	36
• Message Handling	39
• Class UndefinedObject	40
• Debugging	41
• halt	41
<i>Lecture 12: Designing and implementing classes</i>	43
• Steps to develop a specification	43
• The message protocol	43
• Steps to implementing a class	43
• Describing a class	43
<i>Lecture 13: VisualWorks</i>	44
Starting VisualWorks	44
VisualWorks Launcher	44
Workspace	45
Using the Mouse and the Pop-Up Menus	45
Setting up VisualWorks	45
Online Documentation	47
System Browser	47
Filing In and Filing Out Components	48
Filing In	48
Filing Out	49
Starting an Application	49
Saving Your Work	50

<i>Lecture 14: More on the Basic VisualWorks Environment</i>	51
Workspaces	51
The Transcript	52
Editing in VisualWorks Windows	52
Using a Browser	52
Adding a New Method	55
Adding New Classes or Methods From External Files	56
Changing Existing Methods	58
Adding a New Class	58
Saving Code into a File	61
<i>Lecture 15: System & Magnitude Classes</i>	62
• Overview	62
• Shared Object Protocols	62
• 4 basic subclasses of the Magnitude class	63
• Methods provided for comparison	63
• Example: More methods for complex numbers	64
• Partial Hierarchy	64
• Type Conversion	65
• Truncation, floor, ceiling and remainders	67
• Mathematical Operations	67
• Date and Time	68
<i>Lecture 16: The Collection Classes</i>	69
• Smalltalk's optimized Collection classes	69
• Partial Hierarchy	70
• Iteration (what you can do with collections)	70
<i>Lecture 17: An example using the Collection Classes</i>	73
<i>Lecture 18: The Stream Classes</i>	82
• Streams	82
• Important methods for all Streams	82
• Important methods for Positionable Streams	83
• Important methods for ReadStreams	84
• Important methods for WriteStreams	84
• Important methods for External and File Streams	84
• Common Mistakes	85

• Hierarchy	86
<i>Lecture 19: Matrix Example using Streams</i>	87
<i>Lecture 20: Dependency Mechanisms</i>	91
• Dependency	91
<i>Lecture 21: The Model-View-Controller Paradigm</i>	97
• Definitions	97
Model	101
<i>Lecture 22: The View</i>	105
View	105
<i>Lecture 23: The Controller</i>	110
<i>Appendix1: VisualWorks 2.5 versus Smalltalk-80</i>	113
Differences found throughout the lecture note's examples	113
• Classes removed from VisualWorks 2.5	113
<i>Appendix2: VisualWorks rules and Smalltalk Syntax</i>	114
• Capitalization rules	114
• Reserved words	114
• Operators	114
• Literals	114
• Comments	114
<i>Appendix 3: A List of Methods for the System Classes</i>	115
Magnitude:	115
Collection	119
Stream	124
<i>Index</i>	127

Index

abs.....	64	Date.....	63
Abstraction	16, 22	dateAndTimeNow.....	68
addDeponent.....	91	Debugging	41
anAspect.....	91	deepCopy.....	27
and.....	34	Delay.....	109
arcCos.....	67	Dependency	91
arcSin.....	67	dependents.....	91
ArithmeticValue.....	63	DependentsFields.....	95
asCharacter.....	66	detect.....	71
asFloat.....	66	detect:.....	102
asFraction.....	66	Dialog.....	111
asInteger.....	66	Dictionary	70
Associated Hashtable	70	displayObject.....	106, 107
asString.....	14	displayOn:.....	105
at.....	13, 28	do.....	70
AutoScrollingView.....	106	doesNotUnderstand.....	36
basicAt.....	29	Encapsulation	20
basicSize.....	29	equivalence.....	27
Behavior	20	eqv.....	34
Binary	30	error.....	37
BlockClosure.....	34	Error Handling	36
Blocks	34	errorSignal.....	41
blueButtonPressed.....	110	Exception	41
Boolean	34	exp.....	67
Branching	34	Exponents.....	67
Browser	52	Factorization	18, 22
bytecode.....	14	false.....	114
Capitalization.....	114	False.....	34
ceiling.....	67	File Streams	84
changed.....	91	Filing In	48
Char.....	63	Filing Out	48
class.....	27	findFirst.....	71
Class	11	findLast.....	71
Class Hierarchy	11	floor.....	67
Class instance variables	32	flush.....	84
Class Protocol.....	43	Global Variables	33
Class Variables	33	GraphicsContext.....	106
closeAndUnschedule.....	111	halt	41
collect.....	41, 71	haltSignal.....	41
Collection.....	62	hardhalt.....	41
Comments.....	114	hash.....	63
Comparison	27	ifFalse.....	34, 35
Composition	17, 22	ifTrue.....	34
confirm.....	41	image.....	14
confirm:.....	110	inject.....	71
Control Structures	<i>See</i> Branching	inspect.....	41
controlActivity.....	110	Instance	11
Controller.....	97, 110	Instance Protocol.....	43
copy.....	27	Instance variables	32
Copying objects	27	InstVarAt.....	40
cos.....	67	isKindOf.....	27

isMemberOf.....	27
isNil.....	40, 62
isSequenceable.....	62
Iteration	70
keyboardPressed.....	110
Keyword	30
Launcher	44
Literals	114
ln.....	67
logarithms.....	67
Magnitude.....	62
max.....	63
message protocol	43
messageNotUnderstoodSignal.....	41
Messages	13, 14, 30
Method arguments	32
Method Lookup	30
Methods	13, 30
min.....	63
Model	97, 101
newDay.....	68
next.....	82
nextNumber.....	84
nextPut.....	82, 84
nextPutAll.....	83, 85
nextString.....	84
nil.....	40, 114
notify.....	41
notNil.....	40, 62
now.....	68
Object	7
Operators	114
or.....	34
peek.....	83
perform.....	39
Polymorphism	21
position.....	83
Positionable Streams	83
primitiveFailed.....	37
printString.....	35, 65
Procedural Approach	7
quo.....	67
readFrom.....	66
readFromString.....	29
ReadStream	84
redButtonPressed.....	110
reject.....	71
release.....	91
rem.....	<i>See remainder</i>
remainder.....	67
removeDependent.....	91
reset.....	83
respondsTo.....	27
respondsToArithmetic.....	62
Return Values	33
reverseDo.....	70
ScheduledController,.....	108
ScheduledWindow.....	108
select.....	71
SelectionView	98, 105
self.....	35, 114
SequencableCollecection.....	62
SequenceableCollecection.....	62
shallowCopy.....	28
shouldNotImplement.....	37
signal	41
sin.....	67
size.....	29
skip.....	84
skipwords.....	84
Specialization	15
stdin.....	82
stdout	82
storeString.....	65
Streams	62, 82
subclassResponsibility.....	38
subclassResponsibilitySignal.....	41
super.....	114
System Browser	47
Temp variables	32
TextCollectorView	105
thisContext.....	114
Time.....	63
timesRepeat.....	35
today.....	68
Transcript	52
Trigonometry	67
true.....	114
True.....	34
Truncation.....	67
Unary.....	30
UndefinedObject	40
update.....	91
update:.....	91, 101
Variables	32
View	97, 105
Virtual Machine	14
whileFalse.....	35
whileTrue.....	35
WindowSensor	110
with: do.....	71
wordPosition.....	84
Workspace	45, 51
WriteStreams	84
xor.....	34
year.....	68

yellowButtonPressed.....110

Yourself.....29

