Appendix A

C/C++

Notes

Operator Precedence

NOTE A.1

Operator Precedence

Default operation	***C and C++***	***C++ Only***	***Associativity***
scope		::	left to right
primary	( ) [ ] −> .	*type*( ) const_cast dynamic_cast reinterpret_cast static_cast typeid	left to right
unary	++ −− ! ~ (*type*) + − ∗ & sizeof	new delete	right to left
select pointer		.* −>*	left to right
multiplicative	* / %		left to right
additive	+ −		left to right
shift	<< >>		left to right
relational	< <= > >=		left to right
equality	== !=		left to right
bitwise	&		left to right
bitwise	^		left to right
bitwise	|		left to right
logical	&&		left to right
logical	||		left to right
conditional	?:		right to left
assignment	= += −= *= /= %= <<= >>= &= |= ^=		right to left
throw		throw	left to right
comma	,		left to right

NOTE A.2

Default Operator Meanings

Operators Common to Both C and C++					
( )	Function Call	[ ]	Array access	–>	Struct/Union Ptr
.	Struct/Union Mbr	++	Increment	- -	Decrement
!	Logical Negation	~	One's Complement	(*type*)	Typecast
+	Unary Plus	-	Unary Minus	*	Indirection
&	Address	sizeof	Byte Count	*	Multiplication
/	Division	%	Modulus	+	Addition
-	Subtraction	<<	Left Shift	>>	Right Shift
<	Less Than	<=	Less or Equal	>	Greater Than
>=	Greater or Equal	==	Equality	!=	Inequality
&	Bitwise And	^	Exclusive Or	|	Bitwise Or
&&	Logical And	||	Logical Or	?:	Conditional
= += -= *= /= %= <<= >>= &= |= ^=			Assignment	,	Comma

C++ Only Operators			
::	Scope Resolution	typeid	Type Identification
type( )	Typecast	new	Allocate Memory
const_cast	Typecast	delete	Deallocate Memory
dynamic_cast	Typecast	.*	Member Dereference
reinterpret_cast	Typecast	–>*	Indirect Member Dereference
static_cast	Typecast	throw	Throw Exception