

Control Structures

Control Statements provide us flexibility to control the flow of execution

Logical (Conditional) AND, OR, and NOT

Logical operators applications

Nested if's

Objectives and Learning Outcomes

- Learn Logical operators
- Learn Applications of Use primitive types and operators

Logical Operators

- Boolean expressions can use logical (conditional) Statements Which are Boolean
- Result of these expression is Boolean(True,False)

!	Logical NOT(Unary Operator)	
&&	Logical AND	} (Binary Operator)
	Logical OR	

- They all take boolean operands and produce boolean results

Symbol	Meaning	Example
&&	conditional AND	(a > 20) && (a < 35)
	conditional OR	(height > 78.5) (height < 300)

age	age > 20	age < 35	age > 20 && age < 35
10	false	true	false
25	true	true	true
40	true	false	false

Num1	Num2	C1= Num1 > Num2	C2= Num2 > 250	C1 C2
62	125	false	false	false
80	25	false	True	True
72	310	false	True	True
480	325	True	True	True

Short Circuit Property of && and ||

A	B	A && B
true	true	true
true	false	false
false	(don't care)	false

A	B	A B
true	(don't care)	true
false	true	true
false	false	false

A	!A
true	false
false	true