Class 05

Repeated Decisions, Logical Operators &&, II, ! (and, or, not)

Repeated Decisions - Loops

if (true/false question) {

statement(s); // executes only if question is true

}

If question is true, statement is executed and program moves to the next line after the if statement

while (true/false question) {

statement(s); // executes repeatedly as long as question is true

}

If question is true, statement executes repeatedly until the question becomes false

Example Program – funClass.cpp

- Plan:
  - Declare char variable ans and initialize to ‘y’
  - In a while loop, ask the user if they are having fun in class
  - As long as the user enters “y” continue asking the question
  - If the user does not enter “y” tell them “That’s too bad.”

Boolean Operators

Operator	Meaning	Example	Result
&&	Logical And	(5 < 2) && (5 > 3)	FALSE
||	Logical Or	(5 < 2) || (5 > 3)	TRUE
!	Logical Not	! (5 < 2)	TRUE

Example – password.cpp

- Goal: Ask for a three-digit password
- If the password is illegal, keep asking again and again until it is legal

Examples of conditions (single and compound)

- Examples using count = 0, limit = 10, x = 12, y = 15
  - (x == 12)
  - !(count != limit)
  - (count < 10) && (x < y)
  - (limit < 20) || ((limit / x) > 7)
  - (count == 0) && (x > y)
  - !(x == count)
  - !((limit != x) || (count < limit))

Example Program – funClass2.cpp

- Plan:
  - Declare char variables ans1 and ans2 and initialize each to ‘y’
  - In a while loop:
    - ask the user if they are having fun in class
    - ask the user if they are lying right now
  - As long as the user enters “y” for both continue asking the question
  - If the user is not having fun in class, print “That’s too bad.”
  - If the user is lying right now, print “Truth hurts.”