

Conditions and Logical Expressions

C Programming

Lecture Topics

- **Using Relational and Logical Operators to Construct and Evaluate Logical Expressions**
- ***If-Else* Statements**

Flow of Control

- is **Sequential** unless a “control structure” is used to change that
- there are 2 general types of control structures:
 - Selection** (also called branching)
 - Repetition** (also called looping)

C control structures

- **Selection**

 - if**

 - if . . . else**

 - switch**

- **Repetition**

 - for loop**

 - while loop**

 - do . . . while loop**

Control Structures

- C allows a program to make a decision based on the value of a condition. Such a condition must evaluate to true or false.

use logical expressions which may include:

6 Relational Operators

< <= > >= == !=

3 Logical Operators

! && ||

6 Relational Operators

are used in expressions of form:

ExpressionA *Operator* *ExpressionB*

temperature > humidity

$B * B - 4.0 * A * C$ > 0.0

abs (number) == 35

initial != 'Q'

```
int x, y ;
```

```
x = 4;
```

```
y = 6;
```

<u>EXPRESSION</u>	<u>VALUE</u>
$x < y$	true
$x + 2 < y$	false
$x \neq y$	true
$x + 3 \geq y$	true
$y == x$	false
$y == x + 2$	true

Operator

Meaning

Associativity

!	NOT	Right
*, / , %	Multiplication, Division, Modulus	Left
+ , -	Addition, Subtraction	Left
<	Less than	Left
<=	Less than or equal to	Left
>	Greater than	Left
>=	Greater than or equal to	Left
==	Is equal to	Left
!=	Is not equal to	Left
&&	AND	Left
 	OR	Left
=	Assignment	

LOGICAL

EXPRESSION

MEANING

DESCRIPTION

! p	NOT p	! p is false if p is true ! p is true if p is false
p && q	p AND q	p && q is true if both p and q are true. It is false otherwise.
p q	p OR q	p q is true if either p or q or both are true. It is false otherwise.

What is the value?

```
int age, height;
```

```
age = 25;
```

```
height = 70;
```

EXPRESSION

VALUE

!(age < 10)

?

!(height > 60)

?

Short-Circuit Example

```
int age, height;
```

```
age = 25;
```

```
height = 70;
```

EXPRESSION

```
(age > 50) && (height > 60)
```

false

Evaluation can stop now because result of && is only true when both sides are true. It is already determined that the entire expression will be false.

More Short-Circuiting

```
int age, height;  
  
age = 25;  
height = 70;
```

EXPRESSION

```
(height > 60) || (age > 40)
```

true

Evaluation can stop now because result of `||` is true if one side is true. It is already determined that the entire expression will be true.

What happens?

```
int age, weight;  
age = 25;  
weight = 145;
```

EXPRESSION

(weight < 180) && (age >= 20)

true

Must still be evaluated because truth value of entire expression is not yet known. Why? Result of && is only true if both sides are true.

What happens?

```
int age, height;  
age = 25;  
height = 70;
```

EXPRESSION

! (height > 60) || (age > 50)

true

false

Does this part need to be evaluated?

Write an expression for each

taxRate is over 25% and income is less than 20000
(taxRate > .25) && (income < 20000)

temperature is less than or equal to 25 or humidity
is less than 70%
(temperature <= 25) || (humidity < .70)

age is over 21 and age is less than 60
(age > 21) && (age < 60)

age is 21 or 22
(age == 21) || (age == 22)

Comparing float Values

- do not compare float values for equality, compare them for **near-equality**.

```
float myNumber;  
float yourNumber;
```

.....

```
if ( fabs (myNumber - yourNumber) < 0.00001 )  
    printf("They are close enough") ;
```


Understanding Truth In C

the value 0 represents false

ANY non-zero value represents true
(Usually 1)

- For example:
 - -1 is true
 - 299 is true
 - 0 is false

Equality Vs Assignment

- **Given:**
if (grade = 100)
 printf (“Perfect Score!”);
- **This statement does the following:**
 - **Grade is assigned the value 100.**
 - **Because 100 is true (i.e. non-zero!), the condition is always true.**
- **No matter what the student grade, it always says “Perfect Score!”**

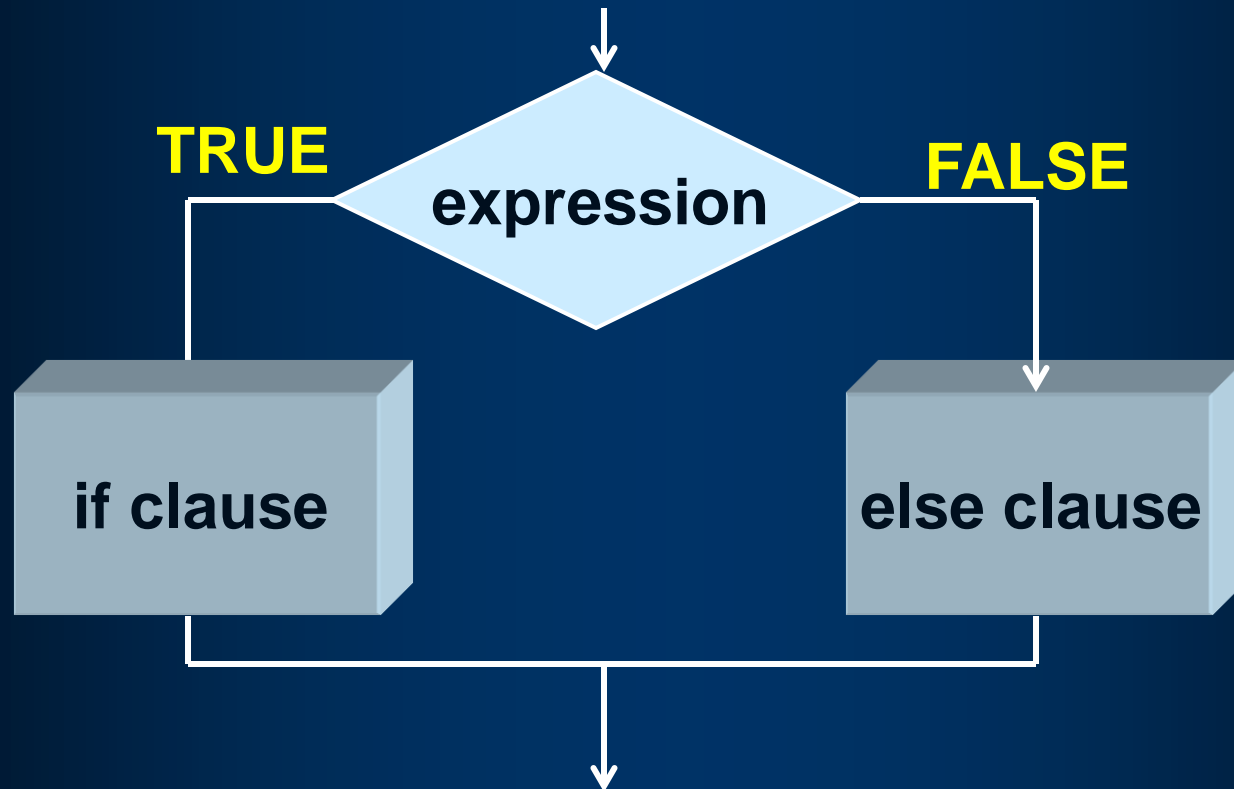
If-Else Syntax

```
if ( Expression )  
    StatementA  
else  
    StatementB
```

NOTE: StatementA and StatementB each can be a single statement, a null statement, or a block.

if - else provides two-way selection

between executing one of 2 clauses (the if clause or the else clause)



Use of blocks recommended

```
if ( Expression )
```

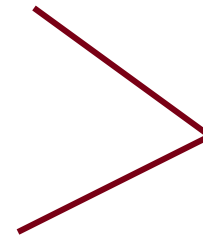
```
{
```

```
}
```

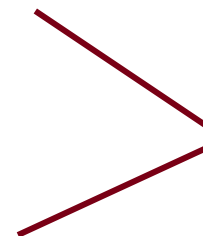
```
else
```

```
{
```

```
}
```



“if clause”



“else clause”

```
int    carDoors, driverAge ;
double premium, monthlyPayment ;
. . .
if ( (carDoors == 4 ) && (driverAge > 24) )
{
    premium = 650.00 ;
    printf( " LOW RISK " ) ;
}
else
{
    premium = 1200.00 ;
    printf("HIGH RISK ") ;
}

monthlyPayment = premium / 12.0 + 5.00 ;
```

What happens if you omit braces?

```
if ( (carDoors == 4 ) && (driverAge > 24) )  
    premium = 650.00 ;  
    printf( " LOW RISK " ) ;  
else  
    premium = 1200.00 ;  
    printf( " HIGH RISK " ) ;  
  
monthlyPayment = premium / 12.0 + 5.00 ;
```

COMPILE ERROR OCCURS. The “if clause” is the single statement following the if.

Braces can only be omitted when each clause is a single statement

```
if ( lastInitial <= 'K' )  
    volume = 1;  
else  
    volume = 2;  
  
printf( "Look it up in volume # %d of the  
phone book", volume ) ;
```


What output? and Why?

```
int code;  
  
code = 0;  
  
if ( ! code )  
    printf( "Yesterday");  
else  
    printf( "Tomorrow");
```

What output? and Why?

```
int age;
```

```
age = 30;
```

```
if ( age < 18 )
```

```
    printf( "Do you drive?" );
```

```
    printf( "Too young to vote" );
```

What output? and Why?

```
int age;
```

```
age = 20;
```

```
if ( age == 16 )
```

```
{
```

```
    printf( "Did you get driver's license?" ) ;
```

```
}
```

If--Else for a mail order

Write a program to calculate the total price of a certain purchase. There is a discount and shipping cost:

- **The discount rate is 25% and the shipping is 10.00 if purchase is over 100.00.**
- **Otherwise, The discount rate is 15% and the shipping is 5.00 pounds.**

These braces cannot be omitted

```
if ( purchase > 100.00 )
{
    discountRate = .25 ;
    shipCost = 10.00 ;
}
else
{
    discountRate = .15 ;
    shipCost = 5.00 ;
}

totalBill = purchase * (1.0 - discountRate) + shipCost ;
```

Example

- Write a program to ask a student for his grades in 3 exams (each out of 50) , get their total and inform the student whether he passed or failed the course.

Example

The Air Force has asked you to write a program to label aircrafts as military or civilian. Your program input is the plane's speed and its estimated length. For planes traveling faster than 1100 km/hr, you will label those shorter than 52 m "military", and longer as "Civilian". For planes traveling less than 1100, you will issue an "aircraft unknown" statement.