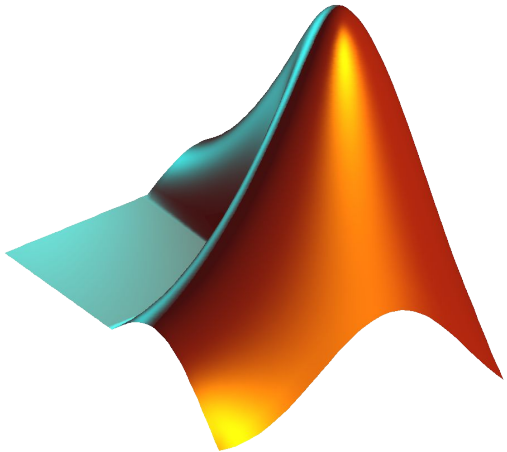
CS 1112 Introduction to Computing Using MATLAB



Instructor: Dominic Diaz

Website: https://www.cs.cornell.edu/courses/cs1112/2022fa/

Today: more on conditionals!

Agenda and announcements

- Last time
  - Tips on writing programs
  - Conditionals (`if`, `elseif`, `else`, `end`)
  - Relational operators (`~=`, `>`, `<=`, `==`)
  - Logical operators (`&&`, `||`, `~`)
- This time
  - More logical operators
  - Nested conditionals
- Announcements
  - Project 1 released (due next Wednesday)
  - Join AEW if you want more practice!
  - Project 1 partner suggestions released on CMS

Random note from last lecture, if you want to print % you would use %%

```
fprintf('You got %d %% on project 1.\n', grade);
```

% would print: You got 90 % on project 1.

The `if` construct

```
if [boolean expression 1]
    [Statements to be executed if expression 1 evaluated to true]
elseif [boolean expression 2]
    [statements to be executed if expression 1 evaluates to false
     but expression 2 evaluates to true]
:
else
    [statements to be executed if all previous expressions
     evaluate to false]
end
```

```
if xc <= L
     fprintf('Increasing\n');
else
     fprintf('Not increasing\n');
end
```

Logical operators

`&&` logical and: are both conditions true?

Example - "is $L \leq x_c$ and $x_c \leq R$?"

In code - `L <= xc && xc <= R`

`||` logical or: is at least one condition true?

Example - "is $x_c \leq L$ or $R \leq x_c$?"

In code - `xc <= L || R <= xc`

`~` logical not: negation

Example - "is x_c not outside [L,R]?"

In code - `~(xc < L || R < xc)`

Poll everywhere question

Write a program that takes in two numbers as inputs and prints the value of the larger one.

```
% Determine which of two numbers are larger

a = input('Enter the first number, a: \n');
b = input('Enter the second number, b: \n');

if ____________________
     fprintf('The larger number is %f.\n', a);
else
     fprintf('The larger number is %f.\n', b);
end
```

a) `a ~= b`

b) `a <= b`

c) `~(a <= b)`

d) `~(a < b)`

Truth table

Let X and Y represent boolean expressions (T/F) [ex: `d > 3.14`]

X	Y	X && Y	X || Y	~Y
F	F			
F	T			
T	F			
T	T			

Truth table

Let X and Y represent boolean expressions (T/F) [ex: `d > 3.14`]

X	Y	X && Y	X || Y	~Y
F	F	F	F	T
F	T	F	T	F
T	F	F	T	T
T	T	T	T	F

MATLAB uses:

0 to represent false

1 to represent true

Truth table

Let X and Y represent boolean expressions (T/F) [ex: `d > 3.14`]

X	Y	X && Y	X || Y	~Y
0	0	0	0	1
0	1	0	1	0
1	0	0	1	1
1	1	1	1	0

For example,

```
X = true;
Y = false;
Z = X || Y;     % Z would store true
Z = ~(X || Y);  % Z would store false
```

Poll everywhere question 2

What is the value stored in Z at the end of this script?

```
X = true;
Y = 0;
Z = X || Y;
Z = ~Z;
```

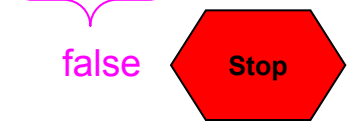
a) true (1)

b) false (0)

Logical operators “short-circuit”

```
a > b && c > d
```


```
a > b && c > d
```



Entire expression is false since the first part is false!

An && expression short-circuits to false if the left expression evaluates to false.

An || expression short-circuits to ___ if the

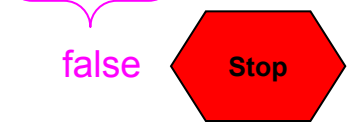
_____

_____

Logical operators “short-circuit”

```
a > b && c > d
```


```
a > b && c > d
```



Entire expression is false since the first part is false!

An `&&` expression short-circuits to false if the left expression evaluates to false.

An `||` expression short-circuits to true if the left expression evaluates to true.

Always use logical operators to connect multiple booleans

If you want to check $L \le x_c \le R$

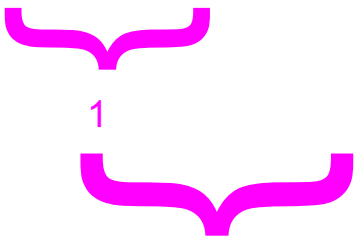
~~`L <= xc <= R`~~

Instead use:

```
L <= xc && xc <= R
```

For example, suppose L = 5, R = 8, xc = 10. The answer should be false.

`L <= xc <= R` gives…

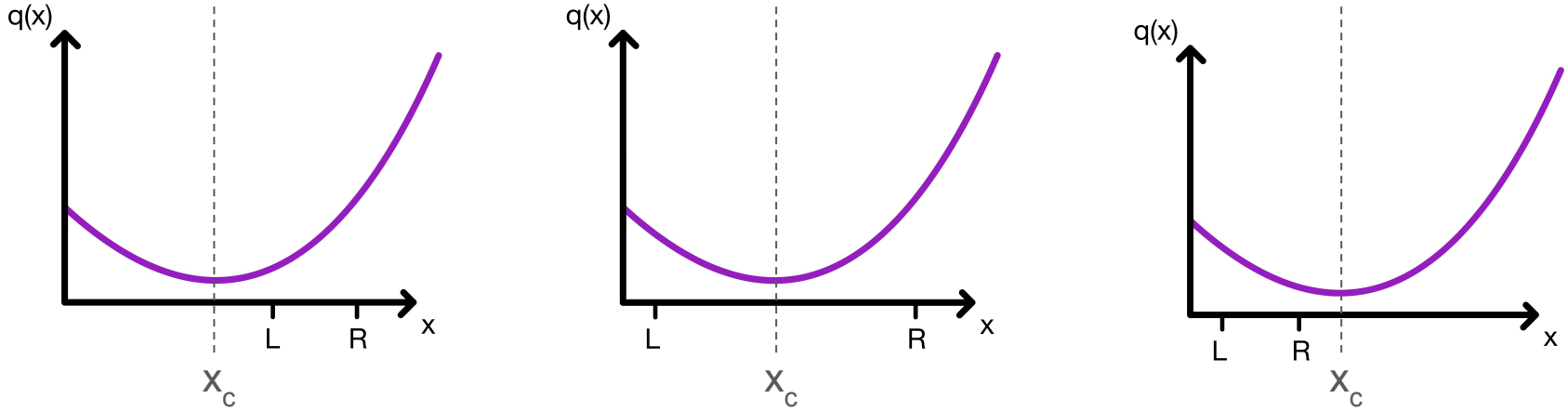


Wrong answer!!!

Variables a, b, and c have whole number values. True or false: This fragment prints “Yes” if there is a right triangle with side lengths a, b, c and prints “No” otherwise.

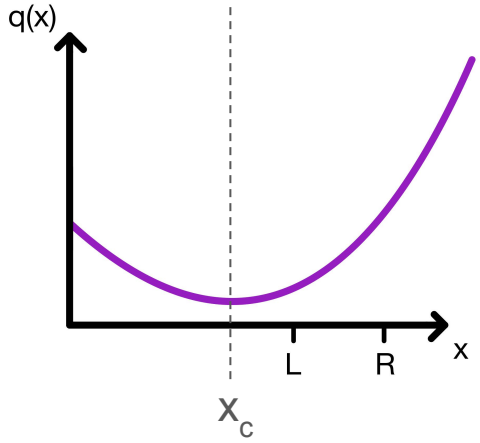
Nested conditionals

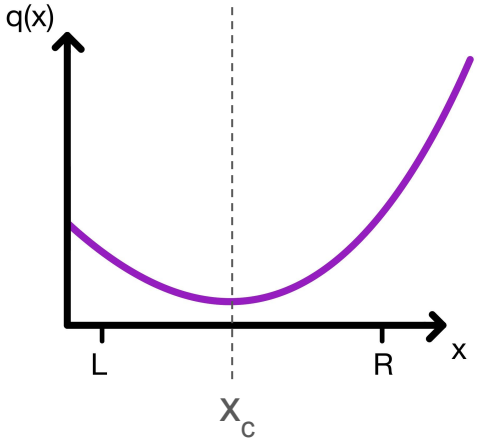
Consider the quadratic function $q(x) = x^2 + bx + c$ on the interval [L, R]. Write a program that prints out the minimum value of q(x) on the interval.

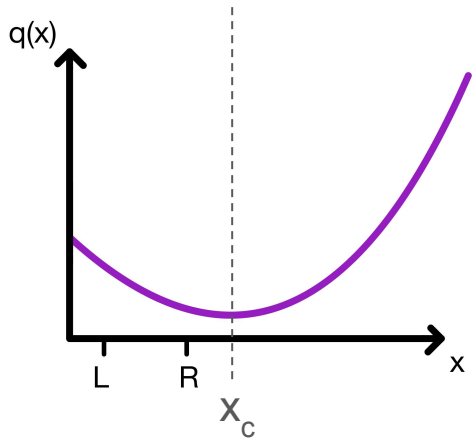


How do we go about solving this problem? First write instructions that a human can understand: pseudocode!

Understand the problem -> write pseudocode







Pseudocode:

If x_c is between L and R,
    Then min is at x_c
Otherwise,
    Then min is at one of the endpoints

We have decomposed the problem into three different parts: the if-else condition, min at xc, and min at an endpoint.

Set up structure first

```
if L <= xc && xc <= R
    % Min is at xc
else
    % Min is at one of the endpoints
end
```

Now that we have the structure, let's refine our solution-in-progress. I would choose to work on the if branch first.

The else statement has two different situations. This calls for another if statement!

Refine your code (nested if-statements)

```
if L <= xc && xc <= R
    qMin = xc^2 + b*xc + c;      % min is at xc
else
    % min is at one of the endpoints
    if % xc is left of interval
        % min is at L
    else % xc is right of interval
        % min is at R
    end
end
```

Continue with refinement; replace comments with code

Filling in the missing blanks

```
if L <= xc && xc <= R
    qMin = xc^2 + b*xc + c;      % min is at xc
else
    % min is at one of the endpoints
    if xc < L
        qMin = L^2 + b*L + c;
    else
        qMin = R^2 + b*R + c;
    end
end
```

For checking multiple criteria, you have many options

```
if L <= xc && xc <= R
    qMin = xc^2 + b*xc + c;
else
    % min at endpoint
    if xc < L
        qMin = L^2 + b*L + c;
    else
        qMin = R^2 + b*R + c;
    end
end
```

```
if L <= xc && xc <= R
    qMin = xc^2 + b*xc + c;
elseif xc < L
    % min at left
    qMin = L^2 + b*L + c;
else
    % min at right
    qMin = R^2 + b*R + c;
end
```