Consider the quadratic function

\[ q(x) = x^2 + bx + c \]

on the interval \([L, R]\):

- Is the function strictly increasing in \([L, R]\)?
- Which is smaller, \(q(L)\) or \(q(R)\)?
- What is the minimum value of \(q(x)\) in \([L, R]\)?

**Modified Problem 3**

Write a code fragment that prints “yes” if \(xc\) is in the interval and “no” if it is not.

```matlab
q(x) = x^2 + bx + c
\bullet \ x_c = -b/2

\begin{align*}
\text{The value of a boolean expression is either true or false.} \\
(L \leq xc) \land (xc \leq R)
\end{align*}

This (compound) boolean expression is made up of two (simple) boolean expressions. Each has a value that is either true or false.

Connect boolean expressions by boolean operators:

\begin{align*}
\text{and} & \quad \text{or} & \quad \text{not} \\
& \quad \land & \quad \lor & \quad \sim
\end{align*}

So what is the requirement?

```matlab
\begin{verbatim}
\% Determine whether xc is in [L,R]
xc = -b/2;
if \_______________
    disp('Yes')
else
    disp('No')
end
\end{verbatim}

% Determine whether xc is in [L,R]
xc = -b/2;
if \_______________
    disp('Yes')
else
    disp('No')
end
Logical operators

&& logical and: Are both conditions true?
E.g., we ask “is \( L \leq x_c \) and \( x_c \leq R \)?”
In our code: \( L \leq x_c \) && \( x_c \leq R \)

|| logical or: Is at least one condition true?
E.g., we can ask if \( x_c \) is outside \([L, R]\), i.e., “is \( x_c < L \) or \( R < x_c \)?”
In code: \( x_c < L \) || \( R < x_c \)

~ logical not: Negation
E.g., we can ask if \( x_c \) is not outside \([L, R]\).
In code: \( \sim (x_c < L \) || \( R < x_c) \)

“Truth table”
X, Y represent boolean expressions.
E.g., \( d > 3.14 \)

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

Logical operators “short-circuit”

\[ a > b \, \&\& \, c > d \]
\[ a > b \, \&\& \, c > d \]

A && expression short-circuits to false if the left operand evaluates to false.
A || expression short-circuits to ____________ if ____________

Entire expression is false since the first part is false

Always use logical operators to connect simple boolean expressions

Why is it wrong to use the expression
\[ L <= x_c <= R \]
for checking if \( x_c \) is in \([L, R]\)?

Example: Suppose \( L = 5 \), \( R = 8 \), and \( x_c = 10 \). We know that 10 is not in \([5, 8]\), but the expression
\[ L <= x_c <= R \] gives...

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 \( a \), \( b \), and \( c \) and prints “No” otherwise.

```matlab
if a^2 + b^2 == c^2
    disp('Yes')
else
    disp('No')
end
```

Consider the quadratic function
\[ q(x) = x^2 + bx + c \]
on the interval \([L, R]\):

*Is the function strictly increasing in \([L, R]\)?*

*Which is smaller, \( q(L) \) or \( q(R) \)?*

*What is the minimum value of \( q(x) \) in \([L, R]\)?*
Set up structure first: if-else, condition

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

Now refine our solution-in-progress. I'll choose to work on the if-branch next.

Refrinement: filled in detail for task “min at xc”

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

Continue with refining the solution... else-branch next

Refrinement: detail for task “min at an endpoint”

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

Continue with the refinement, i.e., replace comments with code.

Final solution (given b,c,L,R,xc)

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

See quadMin.m
quadMinGraph.m

An if statement can appear within a branch—just like any other kind of statement!
Notice that there are 3 alternatives → can use `elseif`!

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

Top-Down Design

- **State problem**
- **Define inputs & outputs**
- **Design algorithm**
  - **Decomposition**
  - **Stepwise refinement**
- **Convert algorithm to program**

An algorithm is an idea. To use an algorithm you must choose a programming language and implement the algorithm.

### Does this program work?

```plaintext
score = input('Enter score: ');
if score > 55
    disp('D')
elseif score > 65
    disp('C')
elseif score > 80
    disp('B')
elseif score > 93
    disp('A')
else
    disp('Not good…')
end
```

A: yes
B: no

### Question

A stick of unit length is split into two pieces. The breakpoint is randomly selected. On average, how long is the shorter piece?

- Physical experiment?
- Thought experiment? → analysis
- Computational experiment! → simulation*

*Need to repeat many trials!

### Question

A stick of unit length is split into two pieces. The breakpoint is randomly selected. On average, how long is the shorter piece?

- A: 0.000001
- B: 0.25
- C: 0.333333
- D: 0.499999
- E: none of the above