**Topics:** Branching (conditional statement)

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

$Q_1$: Does $q(x)$ increase across $[L, R]$?

$Q_2$: Which is smaller, $q(L)$ or $q(R)$?

$Q_3$: What is the minimum value of $q(x)$ in $[L, R]$?

```
% Does q increase across [L,R]?
  xc = -b/2;

  if ____________________
     disp('Yes')
  else
     disp('No')
  end
```

```
% Which is smaller, q(L) or q(R)?
% Fragment 1
  qL= L^2 + b*L + c;  % q(L)
  qR= R^2 + b*R + c;  % q(R)

  ____________________
     disp('qL less than qR')

  ________
     disp('qR less than or equal to qL')

  ________
```

```
% Fragment 2
  qL= L^2 + b*L + c;  % q(L)
  qR= R^2 + b*R + c;  % q(R)

  if ___________________
      disp('qL equals qR')

  ______________________
      disp('qL less than qR')
  else
      disp('qR less than or equal to qL')
  end
```

**Relational Operators**

| Operator | Meaning |
|---|---|
| $<$ | less than |
| $>$ | greater than |
| $<=$ | less than or equal to |
| $>=$ | greater than or equal to |
| $==$ | equal to |
| $\sim=$ | not equal to |

What if you only want to know if $q(L)$ is close to $q(R)$?

```
% Fragment 3
  tol= 1e-9;          % tolerance
  qL= L^2 + b*L + c;  % q(L)
  qR= R^2 + b*R + c;  % q(R)
  if ( abs(qL-qR) < tol )
      disp('qL is close to qR')
  end
```

## Simple `if` construct

**if** *Condition*
  *Statements to execute if the condition is true*
**else**
  *Statements to execute if the condition is false*
**end**

## The even simpler `if` construct

**if** *Condition*
  *Statements to execute if the condition is true*
**end**

## The general `if` construct

**if** *Condition 1*
  *Statements to execute if condition 1 is true*
**elseif** *Condition 2*
  *Statements to execute if condition 1 is false but condition 2 is true*
⋮
**else**
  *Statements to execute if all previous conditions are false*
**end**

## Rules of the `if` construct

- _____ branch of statements is executed
- _____ **else** clause
- _____ **elseif** clauses

Back to the quadratic function $q(x) = x^2 + bx + c$ on the interval $[L, R]$. Determine whether $x_c$ is in $[L, R]$.

```
xc = -b/2;

if _________________
   disp('Yes')
else
   disp('No')
end
```

A *boolean* expression evaluates to either *true* or *false*. Here is an example:

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

A boolean expression can be made up of other (simpler) boolean expressions that are connected by boolean operators: and, or, not

**Logical Operators**

| *expr1* | *expr2* | *expr1* && *expr2* | *expr1* \|\| *expr2* | $\sim$ *expr2* |
|---|---|---|---|---|
| F | F | | | |
| F | T | | | |
| T | F | | | |
| T | T | | | |