

Foundations of R Software

Lecture 20

Basics of Calculations

::::

Conditional Executions – Nested if else if and ifelse

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Control structures in R :

Control statements,

Functions,

Loops.

1. Conditional execution: if()

Syntax

```
if (condition) {execute commands if condition is TRUE}
```

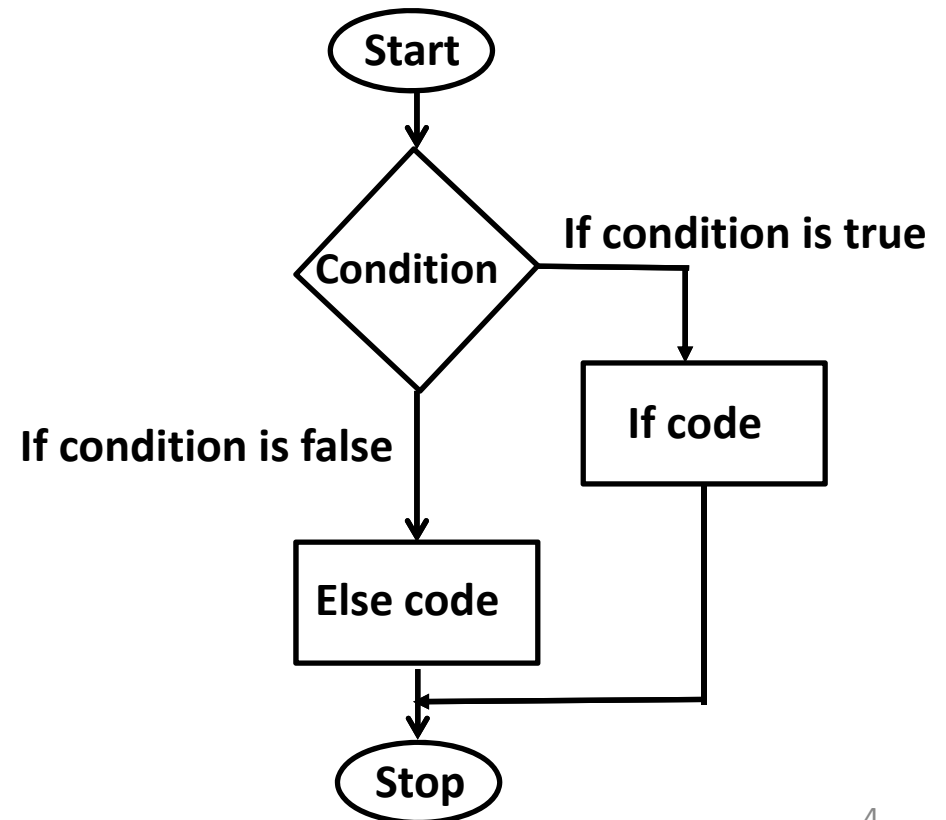
The code in the `{ }` is evaluated if the logical test contained in the `( )` is **TRUE**.

If the logical test is **FALSE**, R will ignore all of the code in the `{ }`.

2. Conditional execution: if else()

Syntax

```
if (condition) {executes commands if condition is TRUE}  
else { executes commands if condition is FALSE }
```



3. Conditional execution: Nested if else if()

The `if...else...if` statement allows to execute a block of code when there are more than two alternatives.

It extends the earlier `if else ( )` condition.

3. Conditional execution: Nested if else if()

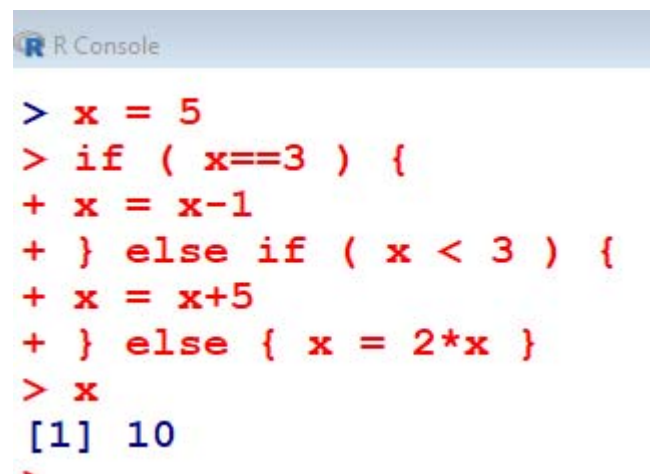
Syntax

```
if (condition1) {  
    executes commands if condition1 is TRUE  
} else if (condition2) {  
    executes commands if condition2 is TRUE  
} else if (condition3) {  
    executes commands if condition3 is TRUE  
}  
... ..  
else {  
    executes commands if all conditions are FALSE  
}
```

3. Conditional execution: Nested if else if()

Example 1:

```
x = 5
if ( x==3 ) {
x = x-1
} else if ( x < 3 ) {
x = x+5
} else { x = 2*x }
x
[1] 10
```

A screenshot of an R console window showing the execution of the code from Example 1. The code is entered line by line, and the output is shown at the bottom.

```
R Console
> x = 5
> if ( x==3 ) {
+ x = x-1
+ } else if ( x < 3 ) {
+ x = x+5
+ } else { x = 2*x }
> x
[1] 10
_
```

Interpretation:

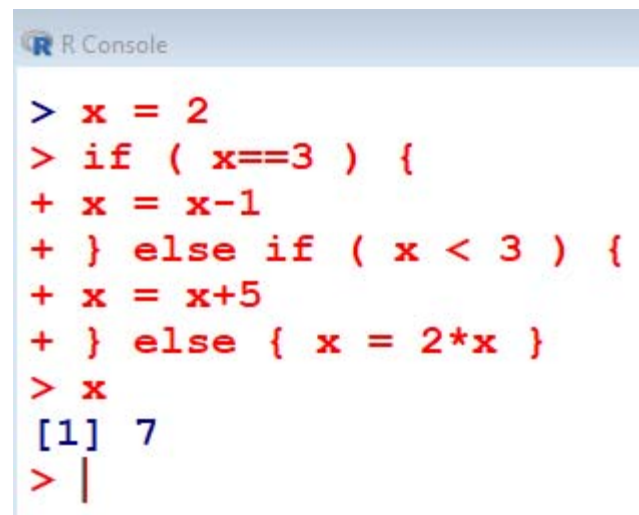
- If $x = 3$, then execute $x = x - 1$.
- If $x < 3$, then execute $x = x + 5$.
- If $x > 3$, then execute $x = 2*x$.

In this case, $x = 5$, so $x > 3$. Thus $x = 2*5$

3. Conditional execution: Nested if else if()

Example 2:

```
x = 2
if ( x==3 ) {
x = x-1
} else if ( x < 3 ) {
x = x+5
} else { x = 2*x }
x
[1] 7
```



```
R Console
> x = 2
> if ( x==3 ) {
+ x = x-1
+ } else if ( x < 3 ) {
+ x = x+5
+ } else { x = 2*x }
> x
[1] 7
> |
```

Interpretation:

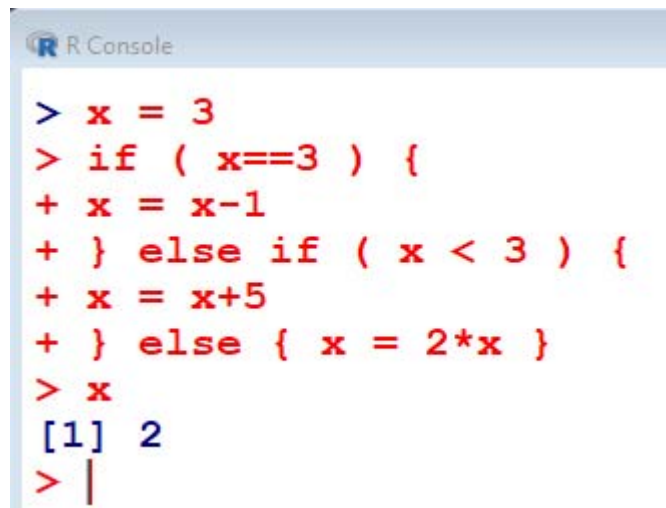
- If $x = 3$, then execute $x = x - 1$.
- If $x < 3$, then execute $x = x + 5$.
- If $x > 3$, then execute $x = 2*x$.

In this case, $x = 2$, so $x < 3$. Thus $x = 2+5$

3. Conditional execution: Nested if else if()

Example 3:

```
x = 3
if ( x==3 ) {
x = x-1
} else if ( x < 3 ) {
x = x+5
} else { x = 2*x }
x
[1] 2
```



```
R Console
> x = 3
> if ( x==3 ) {
+ x = x-1
+ } else if ( x < 3 ) {
+ x = x+5
+ } else { x = 2*x }
> x
[1] 2
> |
```

Interpretation:

- If $x = 3$, then execute $x = x - 1$.
- If $x < 3$, then execute $x = x + 5$.
- If $x > 3$, then execute $x = 2*x$.

In this case, $x = 3$. Thus $x = 3-1$

4. Conditional execution: ifelse()

Syntax

```
ifelse(test, yes, no)
```

- Vector-valued evaluation of conditions .
- For the components in the vector-valued logical expression **test** which provide the value **TRUE**, the operations given by **yes** are executed.
- For the components in the vector-valued logical expression **test** which provide the value **FALSE**, the operations given by **no** are executed.

4. Conditional execution: ifelse()

Example 4:

```
> x = 1:10  
> x  
[1] 1 2 3 4 5 6 7 8 9 10  
> ifelse( x<6, x^2, x+1 )  
[1] 1 4 9 16 25 7 8 9 10 11
```

Interpretation

- If $x < 6$ (TRUE), then $x = x^2$ (YES) .
- If $x \geq 6$ (FALSE), then $x = x + 1$ (NO).
- So for $x = 1, 2, 3, 4, 5$, we get $x = x^2 = 1, 4, 9, 16, 25$
- For $x = 6, 7, 8, 9, 10$, we get $x = x + 1 = 7, 8, 9, 10, 11$

4. Conditional execution: ifelse()

```
R Console  
  
> x = 1:10  
> x  
[1] 1 2 3 4 5 6 7 8 9 10  
> ifelse( x<6, x^2, x+1 )  
[1] 1 4 9 16 25 7 8 9 10 11  
> |
```

4. Conditional execution: ifelse()

Example 5:

```
x = c(7, 9, 8, 4)
```

```
ifelse(x %% 2 == 0, "even number", "odd number")
```

```
[1] "odd number" "odd number" "even number" "even number"
```

[%%: Modulo Division- Finds the remainder after division of one number by another.]

Interpretation

If the remainder of x divided by 2

- is 0, then print "even number" (YES) .
- is not equal to 0, then print "odd number" (NO).

So for x = 7, 9, we get x = "odd number" and

for x=8, 4, we get x= "even number"

4. Conditional execution: ifelse()

Example 5:

R Console

```
> x = c(7,9,8,4)
> ifelse(x %% 2 == 0, "even number", "odd number")
[1] "odd number" "odd number" "even number" "even number"
> |
```