

Control Statements

Conditional statements and loops exist in R; the same as in other programming languages. Given below are the syntax for conditional statements and loops.

```
# If condition if
(test_expression) {
  statement(s)
}

# If...Else condition if
(test_expression) {      statement1
} else {      statement2
}

# Nested If...Else condition if
(test_expression1) {
  statement1 } else if
(test_expression2) {
  statement2
} else if (test_expression3) {
  statement3 } else {      statement4
}
```

```
# while loop while (condition) {
  expressions
}

# for loop
for (symbol in sequence) {
  expressions
}
```

```
}  
for (i in 1:10){  
  print(i)  
} # this loop runs from i = 1 to 10 and prints the value
```

Importing and Exporting

R can read in files on your machine and create data files and graphics. Paths to these files are computed relative to the **working directory**. Paths are specified in the format appropriate for the machine.

R supports basic data types when importing data and other file types can be imported using dedicated packages (e.g.: `xlsx` package for importing Excel 2010 onwards).

`read.table()` can be used to import data from a basic file type and `read.csv()` for importing data from CSV (comma separated values) files. The dataset will be imported as a data frame. The following 2 commands does the same job.

```
> data1 <- read.table(filename, header=TRUE,  
  sep=",")  
> data2 <- read.csv(filename)
```

`write.table()` and `write.csv()` functions can be used to write a data frame to a file.

```
> write.csv(dataframe, file = filename)
```

Functions

Functions are created using the `function()` directive and are stored as R objects just like anything else. In particular, they are R objects of class “function”.

```
fun_name <- function(<arguments>) {  
  statements  
}
```

When specifying arguments, the function can be defined with default values for the arguments.

1. Compute the real roots of the quadratic equation in the form of $ax^2 + bx + c = 0$.
2. Without using R, determine the result of the following computation

```
x <- c(1,2,3)
x[1]/x[2]^3-1+2*x[3]-x[2-1]
```

3. Construct separate plots of $\log(x)$, $\exp(x)$, by using appropriate x values.
4. Consider the vector $1:\mathbb{K}$, where \mathbb{K} is a positive integer. Write an R command that determines how many elements in the vector are exactly divisible by 3.
5. Write an R expression to determine if two sets, A and B , represented as integer vectors are disjoint. If not disjoint, print the common elements.
6. Write a loop structure to scan through an integer vector to determine the index of the maximum value.
7. Do the same without using a loop.
8. Compound interest can be computed using the formula,

$$A = P \times \left(1 + \frac{R}{100}\right)^n$$

where P is the original money lent, A is what it amounts to in n years at R percent per year interest. Write a function to calculate the amount of money owed after n years where n changes from 1 to 15 in yearly increments, if the money lent originally is 5000 rupees and the interest rate remains constant throughout the period at 11.5%.

9. Import the file "Death Row.csv" into R and identify the variables.