Programming Languages

Ritu Arora

Email: ritu@wayne.edu

Date: March 3, 2023

Loops in R

- R has while and for loops

```
i = 0
while (i < 5) {
  print(i)
  i = i + 1
}
```

```
mytasks = list("min", "max", "sort")

for (x in mytasks) {
  print(x)
}
```

- You can use `break` to exit the loop

- You can use `next` to skip an iteration without terminating the loop

- You can have nested loops in R

R Functions

- Functions contain code that is run only when the function is called
- You have to use function() for creating a function in R

```
#creating a function
myFct = function() {
  print(“Hello from R")
}

#calling a function
myFct()
```

R functions with arguments

```
myFct = function(myadject) {
  paste(myadject, "Flower")
}

myFct("Awesome")
myFct("Great")
myFct(“Wonderful")
```

R functions with default values

```
myFct = function(favorite = "Pizza") {
  paste("I like eating", favorite)
}

myFct("Apple")
myFct("Pineapple")
myFct()
```

R functions that return values

```
myFct = function(x) {
  return (x*x)
}

print(myFct(2))
print(myFct(3))
```

Vectors in R

- Vectors are lists of items having the same data type
- The function c() is used to combine the list of items into a vector

```
# Vector containing strings/characters
vector1 = c("hello", "bye", "welcome")

# Print the vector
vector1
```

Sequences in vectors and the length of vectors

```
# Vector with numerical values in a sequence
numbers1 = 2.5:7.5
Numbers1



# Vector with numerical values in a sequence where the last # element is not used
numbers2 = 2.5:7.1
numbers2

# print the length of the vector
length(numbers2)
```

Sorting, updating, and accessing values in vectors

```
#sorting numbers in a sequence
numbers3=c(4, 1, 9, 3, 1)

sort(numbers3)

#Updating the values in the list
numbers[1] = 50
Numbers

#accessing first and fourth value in the list
numbers[c(1, 4)]
```

Repeating

```
#repeating the numbers in the sequence and assigning to a
#new vector
myRepeat = rep(c(4,5,6,7), each = 3)


myRepeat
```

Lists

```
students = c("A","B","C","D")
scores = c(10, 20, 10, 15, 10)
mylist= list(students, scores)
mylist
#check the structure of the list with the str function
str(mylist)
```

Matrix

```
# An array with one dimension with values ranging from 1 to 24
myarray = c(1:24)
myarray


# An array with multiple dimensions - two subarrays with 3 rows
# and 4 columns each
multiarray = array(myarray, dim = c(3, 4, 2))
multiarray
```

Data Frames

```
Data_Frame <- data.frame (
  Students = c("A", "B", "C"),
  Score = c(100.5, 150.5, 120.5),
  Attendance = c(60L, 30L, 45L)
)

# Print the data frame
Data_Frame
```

References

- http://bio.fsu.edu/miller/docs/Tutorials/Tutorial5_IntroProgramming.pdf
- https://www.tacc.utexas.edu/c/document_library/get_file?uuid=2730b001-0036-4c28-9f31-52169dddeb6a&groupId=13601
- http://www.statmethods.net/management/functions.html
- http://math.illinoisstate.edu/dhkim/rstuff/rtutor.html
- https://www.w3schools.com/r/