

Assignment 6

Date: / / 20

Title :- Data Types, Graphics & control structures in R

problem statement :- To study & practice various commands using different data types, Graphics & control structures on R tool & study & practice of various control structures.

Theory

A) Data types in R :

In contrast to other programming language like C & java in R, the variables are not declared as some data types. The variables are assigned with R-objects & data type of R-object becomes data type of variable. There are many types of R-objects. The frequently used ones are -

- Vectors
- list
- Matrices
- Arrays
- factors
- Data frames

The simplest of these objects is vector object & there are six data types of these atomic vectors

Data type	Example .
logical	TRUE, FALSE
Numeric	12.3, 5, 999
Integer	2L, 34L, 0L
Complex	3+2i
Character	'a', "good", "TRUE"
Raw	"Hello" is stored as 48 65 6C 6C 6F .

1) Vectors :-

create vector.

apple <- c ('red', 'green', "yellow")

print (apple)

Get the class of vector.

print (class (apple))

2) List :

create list .

list1 <- list (c(2,5,3), 21.3, sin)

Print the list

print (list1)

3) Matrices .

Create matrix .

M = matrix (c ('a', 'b', 'c', 'b', 'a'), nrow = 2,
n col = 3, byrow = TRUE)

print (M)

4) Arrays :-

```
# create an array.
```

```
a <- array(c('green', 'yellow'), dim = ((3, 3, 2)))
print(a)
```

5) factors :-

```
# create a vector
```

```
apple.colors <- c('green', 'green', 'yellow', 'red', 'red',
                  'red', 'green')
```

```
# create factor object
```

```
factor.apple <- factor(apple.colors)
```

```
# Print factor.
```

```
print(factor.apple)
```

```
print(nlevels(factor.apple))
```

6) Data frames :-

```
# Create data frames
```

```
BMI <- data.frame(gender = c("Male", "male",
                               "female"),
```

```
height = c(152, 171.5, 165),
```

```
weight = c(81, 93, 78),
```

```
Age = c(42, 38, 26)
```

```
)
```

```
print(BMI)
```

7) Strings :-

`b <- "Start & end with double quotes"`
`print(b)`

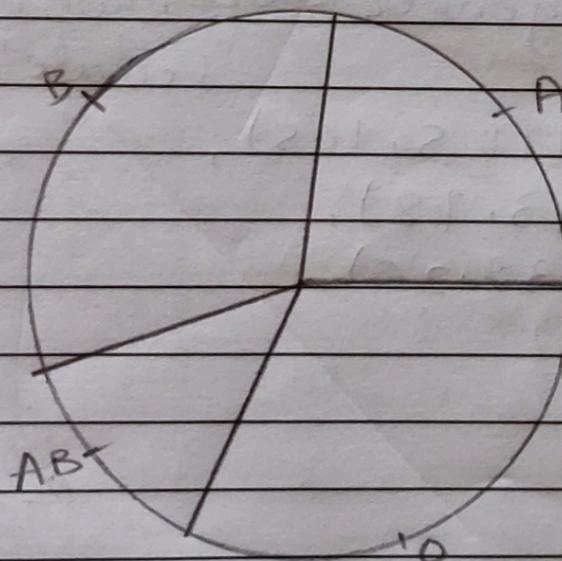
String Manipulation functions used in R:
concatenating Strings - `paste()` function
formatting numbers & strings - `format()`
Counting number of characters in string
`nchar()` function

B. Graphics :-

R programming language has numerous libraries to create charts & graphs.

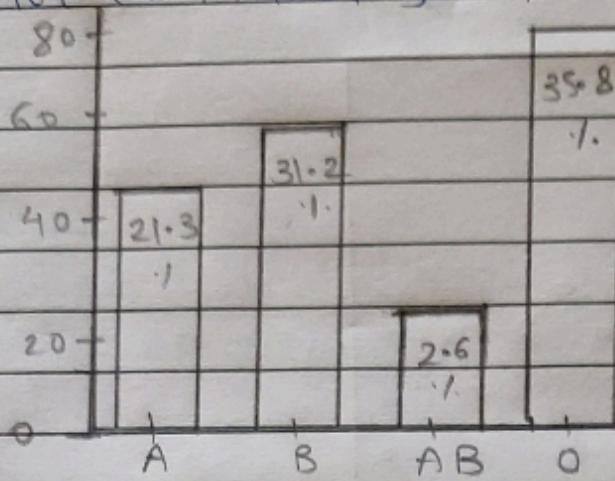
1) Pie chart

The basic syntax for creating pie chart.
`Pie(x, labels, radius, main, col, clockwise)`



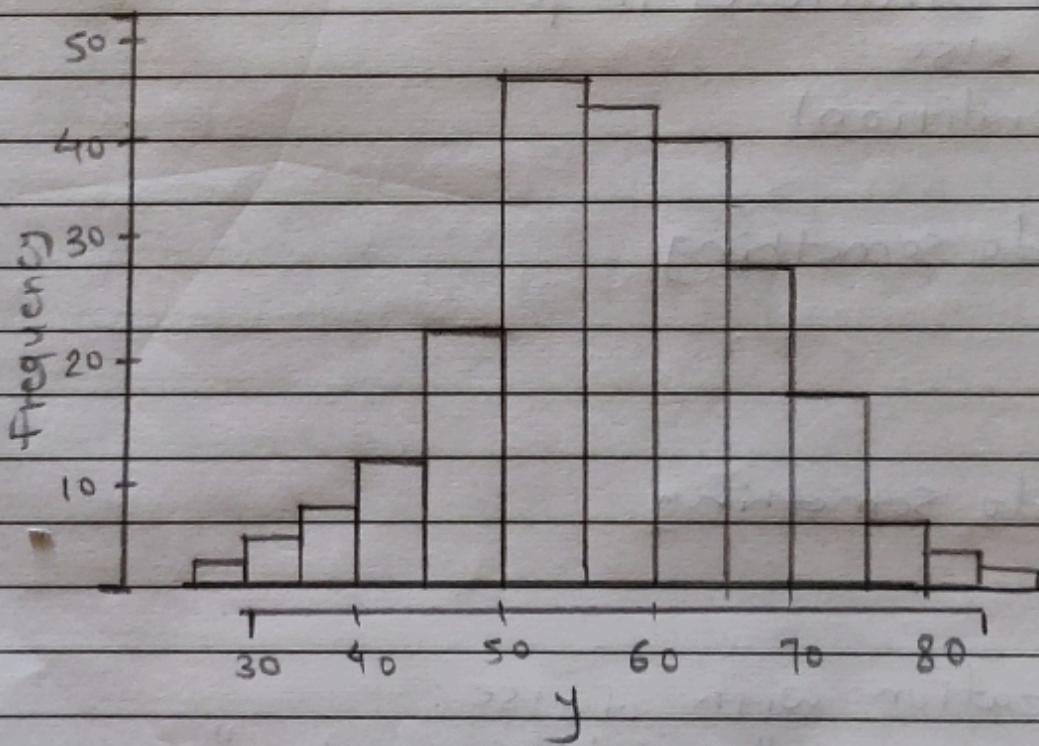
2) Bar Charts -

The basic syntax to create bar-chart is.
`barplot(H, xlab, ylab, main, names.arg, col)`



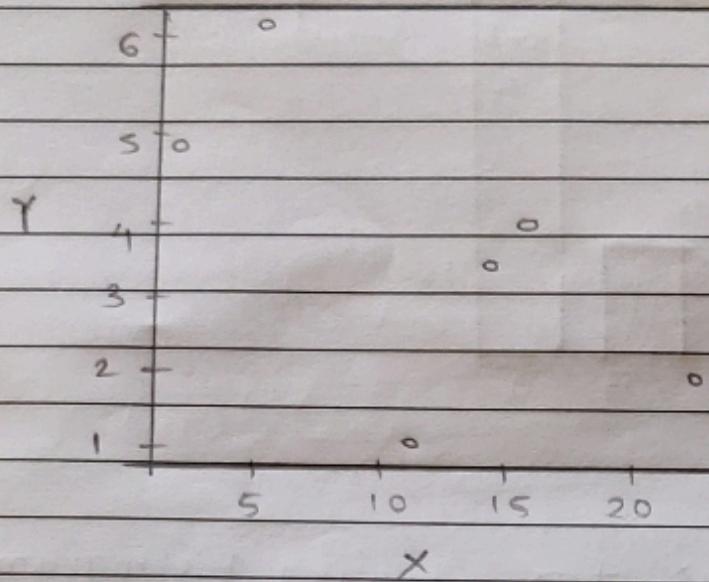
Histograms :

The basic syntax for creating histogram is
`hist(v, main, xlab, xlim, ylim, breaks, col, border)`



Scatterplots :-

The basic syntax for creating scatterplot is
`plot(x, y, main, xlab, ylab, xlim, ylim, axes)`



c) Control Structures in R :

1. If, else :

```
if (condition)
{
    ## do something
}
else
{
    ## do something
}
```

2) vectorization with ifelse :

```
ifelse(x <= 10, "x is less than 10", "x greater than 10")
```

4) For

```
for (<search condition>)
{
    # do something
}
```

5) while .

It begins by testing a condition, & executes
only if condition is found to be true.

 $i \leftarrow 1$

```
while (i < 10)
{
    Print(i)
    i \leftarrow i + 1
}
```

6) Repeat

repeat

{

```
} Statement
```

7) Next .

syntax

next .

8) Break .

Syntax

~~Break~~ break .

Conclusion :- Thus exercised basic syntax, Data types, variables, operators, vectors, lists, matrices, Data frames, factors, Various types of graph & control structures taking suitable examples.