WEEK 1: TUT SHEET 2

Notation: E.g 1:4 in vector/matrix etc means numbers from 1 to 4 i.e. 1, 2, 3, 4.

MATRIX:

- A matrix is a collection of numbers arranged into a fixed number of rows and columns.
- The data elements must be of the same basic type.
- Create Matrix:
  matrix(data, nrow, ncol, byrow, dimnames)
  E.g: Arranging elements by row:

```
matrix_1 <-  matrix(c(1:9),nrow=3, byrow= TRUE)
```

  Arranging elements by column:

```
matrix_2 <-  matrix(c(1:9),nrow=3, byrow= FALSE)
```

  matrix_1: 1 2 3
  4 5 6
  7 8 9

  matrix_2: 1 4 7
  2 5 8
  3 6 9
- Let x and y be two vectors. cbind() combines them by column and rbind() combines them by row.
- To access elements of matrix, matrixName[i][j] should be used , where i is row number and j is column number. matrixName[i][] gives all elements of row i and matrixName[][j] gives all elements of column j.
- To assign names to matrix A having 2 rows and 3 columns.

```
dimnames(A) = list(
c("row1", "row2"),        # row names
c("col1", "col2", "col3")) # column names
```

Exercise: Create three vectors x,y,z with integers and each vector has 3 elements. Combine the three vectors to become a 3×3 matrix A where each column represents a vector.

LISTS:

- R List is the object which contains elements of different types – like strings, numbers, vectors and another list inside it.
- To create a list:
  listName<- list("Hello", "Black", c(1,2,3),4.8, TRUE)
- Accessing list elements:
  E.g. **list_qstp <- list(c("Jan","Feb","Mar"), matrix(c(1,2,3,4,5,6,7,8,9), nrow = 3), list("Hello R",1.3))**

```
list_qstp[1]  #All elements of vector will be accessed
list_qstp[2] #All elements of matrix will be accessed
list_qstp[3] #As it is also a list all its elements will be accessed
```

- Addition, updation, deletion

Addition: list_qstp[4]=”NewElement”
Updation: list_qstp[3]=”UpdateElement”
Deletion: list_qstp[4]=NULL

- Merge two list:
  list1 <- list(1,2,3),”welcome to R”
  list2 <- “Hello”, list(4,5,6)

  list_merge <- c(list1,list2)

Exercise: Write a R program to create a list containing strings, numbers, vectors and a logical values. Access the first and second element of the list.

FOR LOOP AND NESTED FOR LOOP:

a) FOR LOOP:

Syntax:

```
for(i in 1:n)
{
   statement
}
```

Eg:

```
for(i in 1:5)
    {
        print(i^2)
    }
```

b) NESTED FOR LOOP:

Syntax:

```
for(i in 1:n)
{
for(j in 1:n)
{
   Statement
}
}
```

Eg:

```
for(i in 1:5)
    {
    for(j in 1:2)
    {
        print(i*j);
    }
    }
```

BREAK statement:

A **break** is used inside a loop to stop the iterations and flow the control outside of the loop.

Eg:
```
x<-1:5
for(i in x){
    if(i == 3){
    break
    }
    print(i)
}
```
Output:
```
1
2
```

NEXT statement:

Next discontinues a particular iteration and jumps to the next cycle.

Eg:
```
x<-1:5
for(i in x){
    if(i == 3){
    next
    }
    print(i)
}
```
Output:
```
1
2
4
5
```

WHILE LOOP:

Syntax:
```
while(condition){
    statement
}
```

Eg:
```
i <- 1
while (i <=6) {
    print(i*i)
    i = i+1
}
```

Exercise: Write an R program to display cube of all even numbers and square of all odd numbers from 1 to 20.