

# Data Science Practical

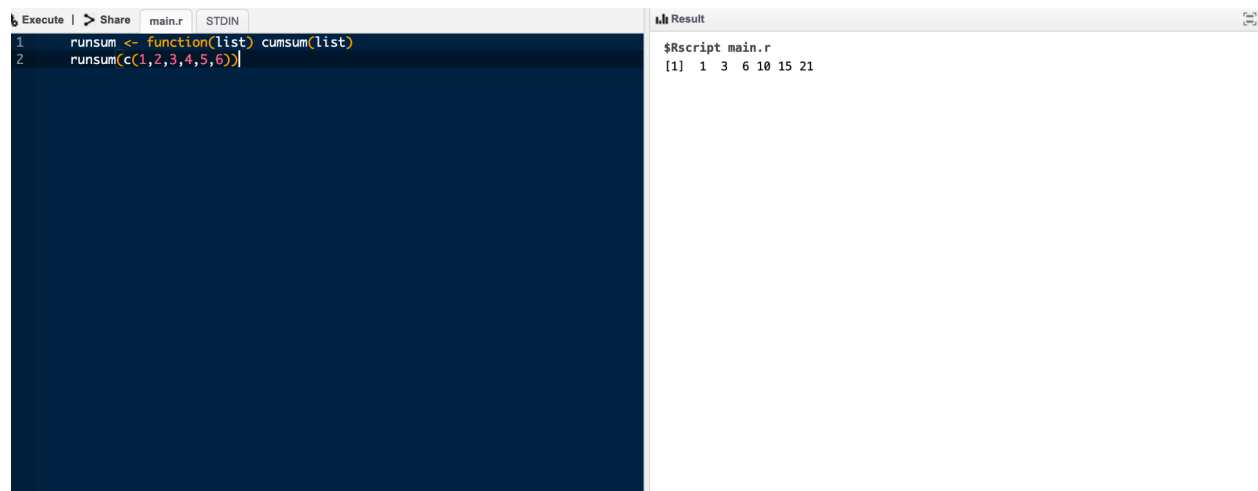
**Name: Shubham Mishra**

**RollNo: 18020570029**

Q1. Write a function that computes the running total of list.

```
runsum <- function(list) cumsum(list)
runsum(c(1,2,3,4,5,6))
```

output:

The screenshot shows an RStudio interface. On the left, the 'Script' editor contains two lines of R code: `1 runsum <- function(list) cumsum(list)` and `2 runsum(c(1,2,3,4,5,6))`. On the right, the 'Console' window shows the output of the second line: `[1] 1 3 6 10 15 21`. The top of the window has tabs for 'Execute', 'Share', 'main.r', and 'STDIN'. The bottom of the window has a 'Result' tab showing the same output: `$Rscript main.r` followed by `[1] 1 3 6 10 15 21`.

Q2. Implement matrices addition, subtraction and Multiplication

```
m1 = matrix(c(1, 2, 3, 4, 5, 6), nrow = 2)
print("Matrix-1:")
print(m1)
m2 = matrix(c(0, 1, 2, 3, 0, 2), nrow = 2)
print("Matrix-2:")
print(m2)

result = m1 + m2
print("Result of addition")
print(result)
```

```

result = m1 - m2
print("Result of subtraction")
print(result)

result = m1 * m2
print("Result of multiplication")
print(result)

```

output:

The screenshot shows an R console window with two panes. The left pane contains the R script, and the right pane shows the output of the script.

**Script (main.r):**

```

1 # Create two 2x3 matrixes.
2 m1 = matrix(c(1, 2, 3, 4, 5, 6), nrow = 2)
3 print("Matrix-1:")
4 print(m1)
5 m2 = matrix(c(0, 1, 2, 3, 0, 2), nrow = 2)
6 print("Matrix-2:")
7 print(m2)
8
9 result = m1 + m2
10 print("Result of addition")
11 print(result)
12
13 result = m1 - m2
14 print("Result of subtraction")
15 print(result)
16
17 result = m1 * m2
18 print("Result of multiplication")
19 print(result)

```

**Result:**

```

$Rscript main.r
[1] "Matrix-1:"
[1,] [,1] [,2] [,3]
[1,] 1 3 5
[2,] 2 4 6
[1] "Matrix-2:"
[1,] [,1] [,2] [,3]
[1,] 0 2 0
[2,] 1 3 2
[1] "Result of addition"
[1,] [,1] [,2] [,3]
[1,] 1 5 5
[2,] 3 7 8
[1] "Result of subtraction"
[1,] [,1] [,2] [,3]
[2,] 1 1 4
[1] "Result of multiplication"
[1,] [,1] [,2] [,3]
[1,] 0 6 0
[2,] 2 12 12

```

### Q3. Implement linear search

```

linSearch <- function(list, element)
{
  pos = 1;
  flag = FALSE;
  for (l in list)
  {
    if (l==element)
    {
      flag = TRUE;
      break;
    }
    pos = pos+1;
  }
  if(flag)
  {
    print(paste("Element found at",pos),quote = FALSE);
  }
}

```

```
    else
    {
        print("Element not found",quote=FALSE);
    }
}

linSearch(c(45,12,1,63,50,12),63)
```

Execute	main.r	STDIN	Result
<pre>1 linSearch &lt;- function(list, element) 2 { 3     pos = 1; 4     flag = FALSE; 5     for (l in list) 6     { 7         if (l==element) 8         { 9             flag = TRUE; 10            break; 11        } 12        pos = pos+1; 13    } 14    if(flag) 15    { 16        print(paste("Element found at",pos),quote = FALSE); 17    } 18    else 19    { 20        print("Element not found",quote=FALSE); 21    } 22 } 23 24 linSearch(c(9,23,51,12),51)</pre>			<pre>\$Rscript main.r [1] Element found at 3</pre>