**Practical - 1**

**Q1. Write a program in Scilab to Calculate the following**

* **Frequency Reuse Distance ( for a Given value of R)**
* **Frequency Reuse Factor**
* **Co-Channel Interference reduction factor (co-channel reuse ratio)**
* **Cellular System Capacity,**
* **S/I Ratio**

**for a given variables. ( Take Default Value of Cluster Size to be 7)**

Ans.

1. Frequency Reuse distance (for a given value of R)

Formula is, **D = R \* SQRT(3\*N)**

**Program code :**

#Frequency reuse distance, D = R\*sqrt(3\*N)

function f = frd()

cluster\_size = 7;

printf("Experiment no 1\n");

printf("Program to calculate the frequency reuse distance\n");

printf("Total number of channels : %d\n", cluster\_size);

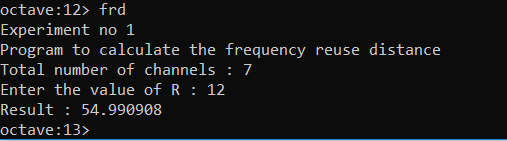
r = input("Enter the value of R : ");

result = r \* sqrt(3\*cluster\_size);

printf("Result : %f\n", result);

endfunction

**Output :**



1. Frequency reuse factor

Formula is, **q = D / R**

**Program code :**

function f = frf()

cluster\_size = 7;

printf("Experiment no 1\n");

printf("Program to calculate the frequency reuse distance\n");

printf("Total number of channels : %d\n", cluster\_size);

r = input("Enter the value of R : ");

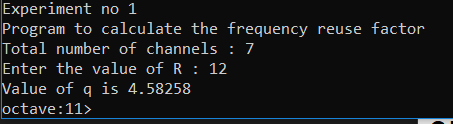
d = r \* sqrt(3\*cluster\_size);

q = d / r;

printf(“Value of q is %d”, q);

endfunction

**Output :**

****

1. Cellular system capacity

Formula is, Capacity = Number of frequencies \* total cluster size

**Program code :**

function f = capacity()

cluster\_size = input("Enter the total clusters in system: ");

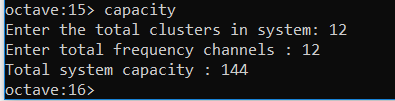
N = input("Enter total frequency channels : ");

capacity = cluster\_size \* N;

printf("Total system capacity : %d\n", capacity);

endfunction

**Output :**

****

1. Signal-interference ratio

Formula is, sir = (sqrt(3\*N)^n)/i0

**Program code**

function f = sir()

num\_cells = input("Enter the number fo cells in thecluster : ");

val\_path\_loss = input("Enter the value of path loss exponent :");

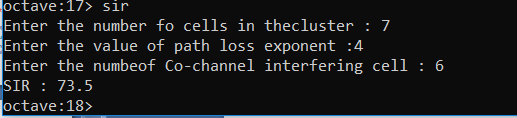
interf\_cell = input("Enter the numbeof Co-channel interfering cell : ");

sir = (sqrt(3\*num\_cells)^ val\_path\_loss)/interf\_cell;

printf("SIR : %d\n", sir);

endfunction

**Output**

****