NAME:	UMANG BHATT
UID:	2021300011
SUBJECT	Design and Analysis of Algorithms
EXPERIMENT NO :	1A
DATE OF PERFORMANCE	24.01.2023
DATE OF SUBMISSION	02.02.2023
AIM:	To implement the various functions e.g., linear, non-linear, quadratic, exponential etc. 1) Print the values of each function value for all n starting 0 to 100 in tabular format for both aforementioned cases 2) Draw two 2D graph of all functions such that x-axis represents the values of n and y-axis represent the function value for different n values using LibreOffice Calc/MS Excel.
THEORY	## What are Functions ? A function is a relation between a set of inputs and a set of permissible outputs with the property that each input is related to exactly one output. Let A & B be any two non-empty sets; mapping from A to B will be a function only when every element in set A has one end, only one image in set B. The 10 functions for analysis are as follows : 1) (3/2)^n 2) n^3 3) ln (ln (n)) 4) log n

	5) $n.2^n$ 6) ln n 7) e^n 8) n 9) 2^n 10) n.(log n) 11) n!
ALGORITHM	Function 1: i. Initialize a variable n of datatype Long Double ii. Take the value of n from 0-100 using for loop iii. n=pow(1.5,i) iv. print all of them. Function 2: i. Initialize variables n and result. ii. Apply a for loop for values of n from 0-100 . iii n=i*i*i* iv. print all the values for n Function 3: i. Initialize variables n . ii. Apply a for loop for values of n from 0-100 . iii. n = ln(ln n) iv. print all the values for n Function 4: i. Initialize variables n. ii. Apply a for loop for values of n from 0-100 . iii. n=log(i) iv. print all the values for n Function 5:

	i. Initialize variables n . ii. Apply a for loop for values of n from 0-100 iii. n = i*2^i iv. print all the values for n. Function 6: i. Initialize variables n. ii. Apply a for loop for values of n from 0-100 iii. n=ln(i) iv. print all the values for n. Function 7: i. Initialize variables n and result. ii. n= pow(e,i) iii. Apply a for loop for values of n from 0-100 iv. print all the values for result. Function 8: i. Initialize variables n. ii. i = n iii. Apply a for loop for values of n from 0-100 iv. print all the values for n. Function 9: i. Initialize variables n and result. ii. n = pow(2,i) iii. Apply a for loop for values of n from 0-100 iv. print all the values for n. Function 10: i. Initialize variables n. ii. n = i*log(i) iii. Apply a for loop for values of n from 0-100 iv. print all the values for n.

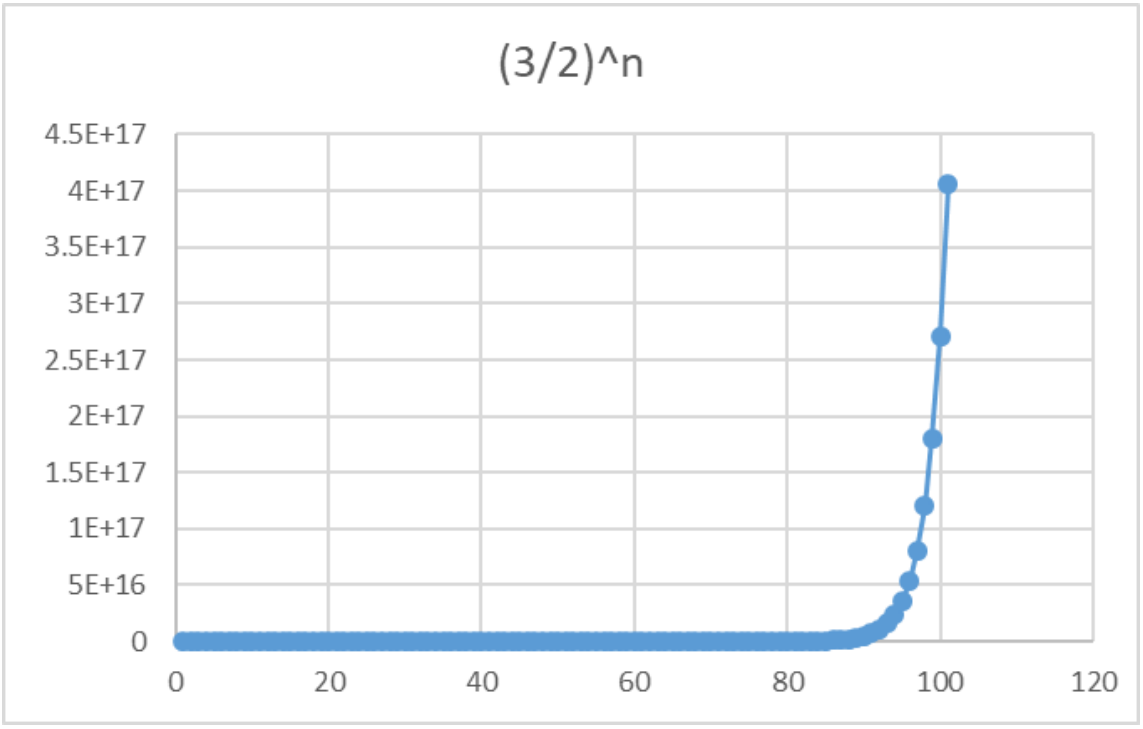
	Function 11:
	i. Initialize a variable n.
	ii. Create a function to find the factorial.
	iii. factorial(n)
	if(n==1 || n==0)
	return i
	else
	return n*factorial(n-1)
	iv. Apply a for loop for values of n from 0-19 and print al the values for result in the main function.

PROGRAM:	#include<stdlib.h>
	#include<conio.h>
	#include<math.h>
	
	int main(){
	int ch;
	printf("the 10 functions are as follows:\n");
	printf("1) (3/2)^n\n2)n^3\n3)ln(ln n)\n4)log n\n5)n*(2^n)\n6)ln n\n7)e^n\n8)n\n9)2^n\n10)n * log n\n");
	ch=1;
	while(ch){
	printf("enter your choice:\n");
	scanf("%d",&ch);
	if(ch==1){
	long double x;
	for(int i=0;i<=100;i++){
	x=pow(1.5,i);
	printf("%.2Lf\n",x);
	}
	}

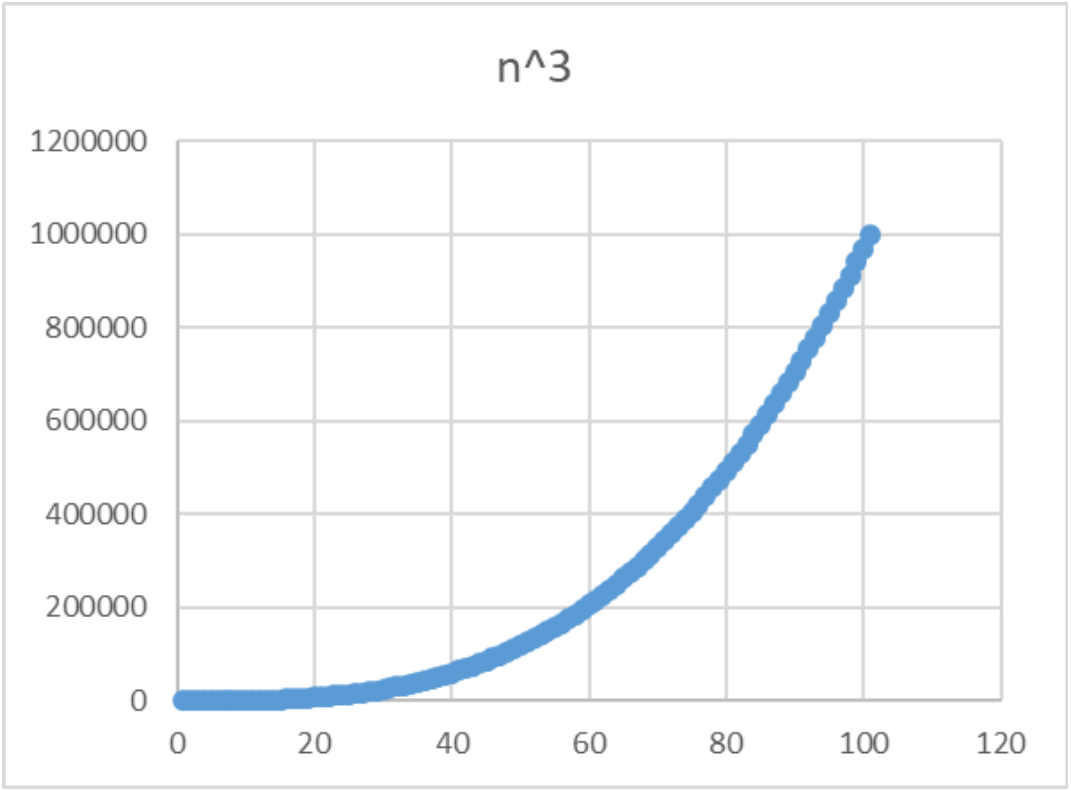
```
    else if(ch==2){
    long double x;
    for(int i=0;i<=100;i++){
        x=pow(i,3);
        printf("%.2Lf\n",x);
    }
}
    else if(ch==3){
    long double x;
    for(int i=0;i<=100;i++){
        x=log(log(i));
        printf("%.2Lf\n",x);
    }
}
    else if(ch==4){
    long double x;
    for(int i=0;i<=100;i++){
        x=log2(i);
        printf("%.2Lf\n",x);
    }
}
    else if(ch==5){
    long double x;
    for(int i=0;i<=100;i++){
        x=i*(pow(2,i));
        printf("%.2Lf\n",x);
    }
}
    else if(ch==6){
    long double x;
    for(int i=0;i<=100;i++){
        x=log(i);
        printf("%.2Lf\n",x);
    }
```

```
        }
            else if(ch==7){
            long double x;
            for(int i=0;i<=100;i++){
                x=pow(2.718,i);
                printf("%.2Lf\n",x);
            }
        }
            else if(ch==8){
            long double x;
            for(int i=0;i<=100;i++){
                x=i;
                printf("%.2Lf\n",x);
            }
        }
            else if(ch==9){
            long double x;
            for(int i=0;i<=100;i++){
                x=pow(2,i);
                printf("%.2Lf\n",x);
            }
        }
            else if(ch==10){
            long double x;
            for(int i=0;i<=100;i++){
                x=i*log(i);
                printf("%.2Lf\n",x);
            }
        }
    }
}
```

RESULT (SNAPSHOT)

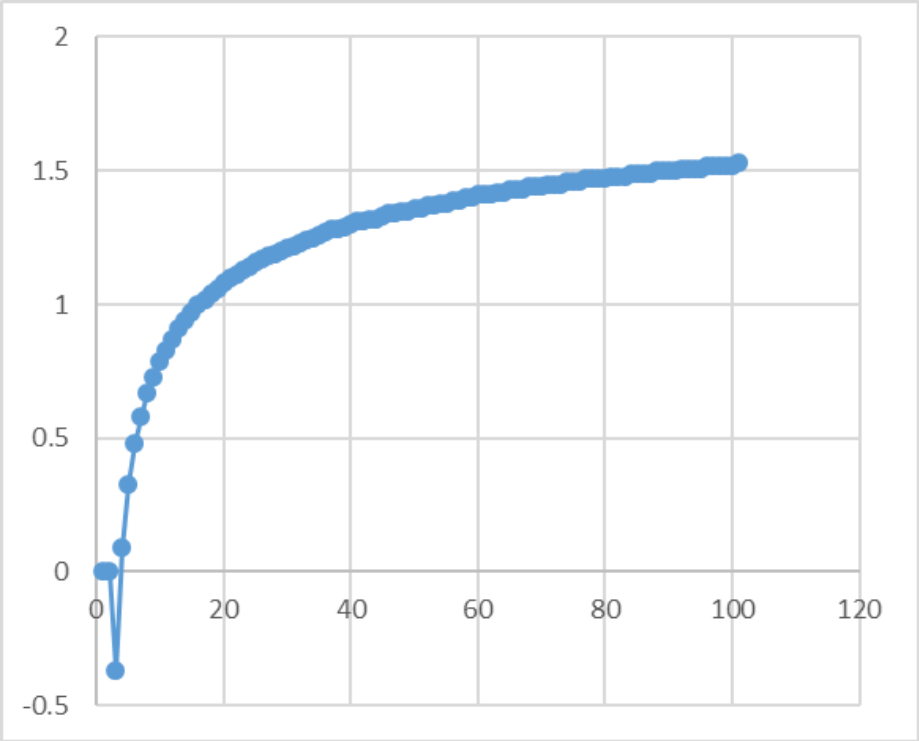


From the above graph we can see that the graph remains almost constant till value of n =90 after which it spikes up drastically between 90 and 100.

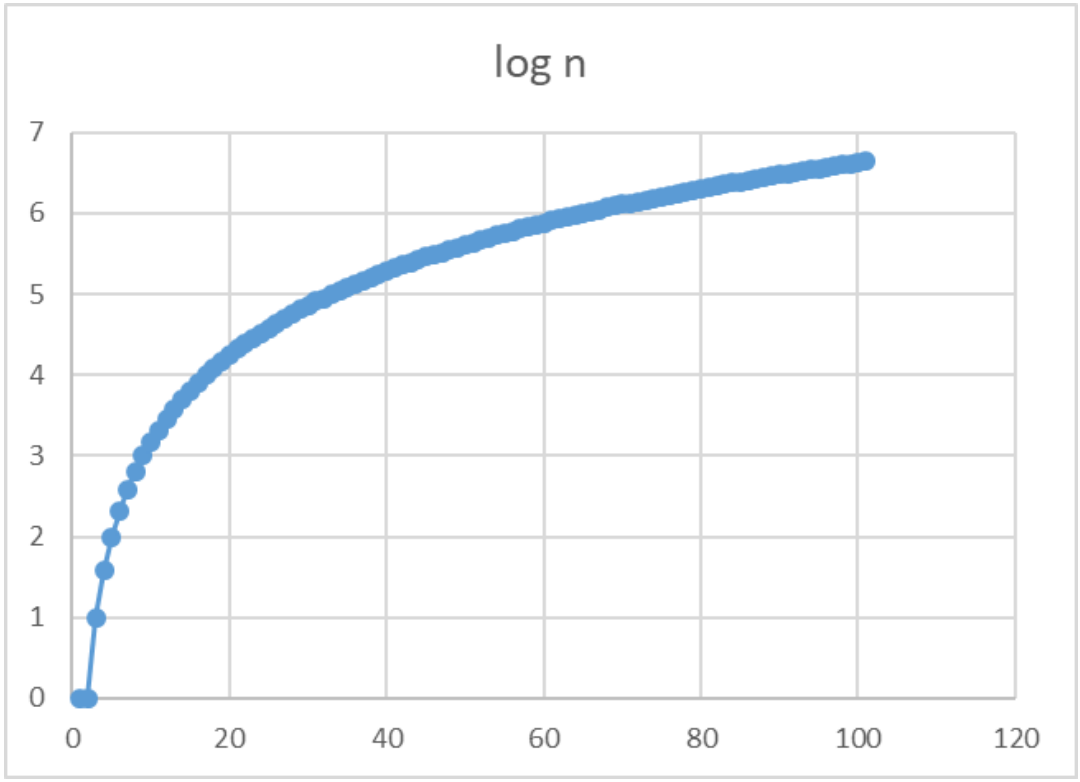


From this graph we observe that, as the value of n increases the graph grows exponentially.

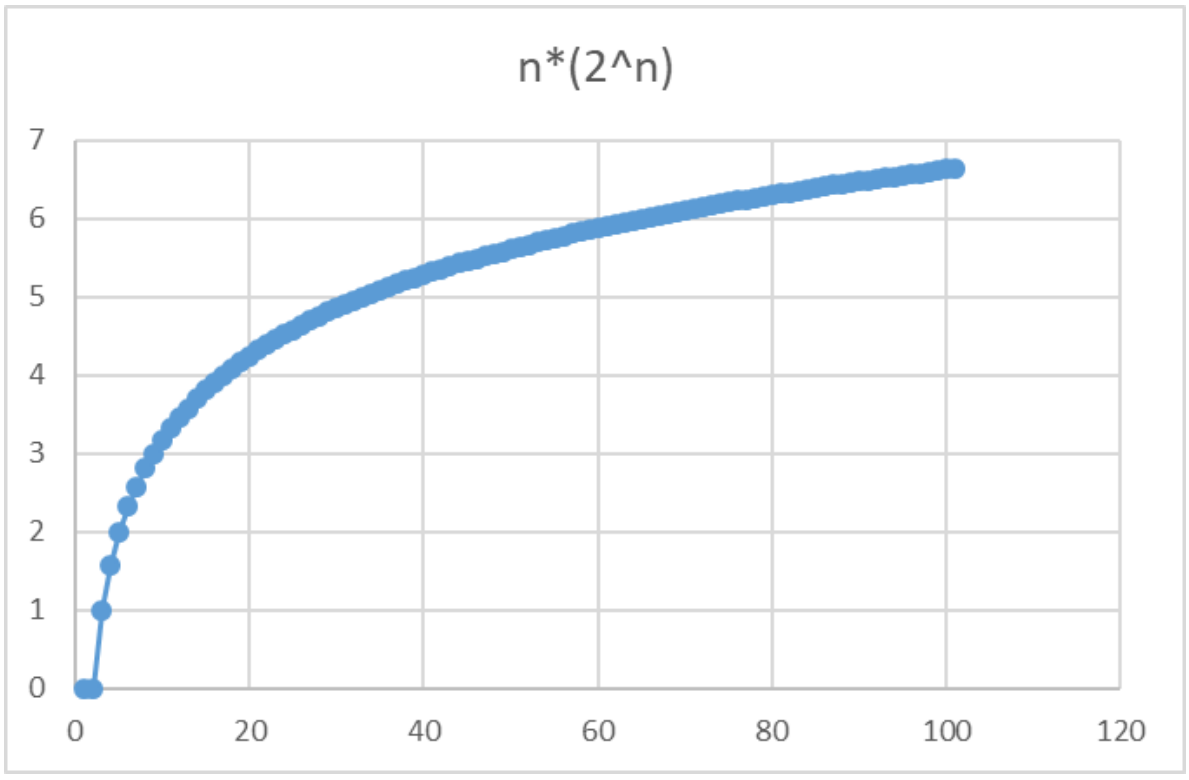
ln(ln (n)):



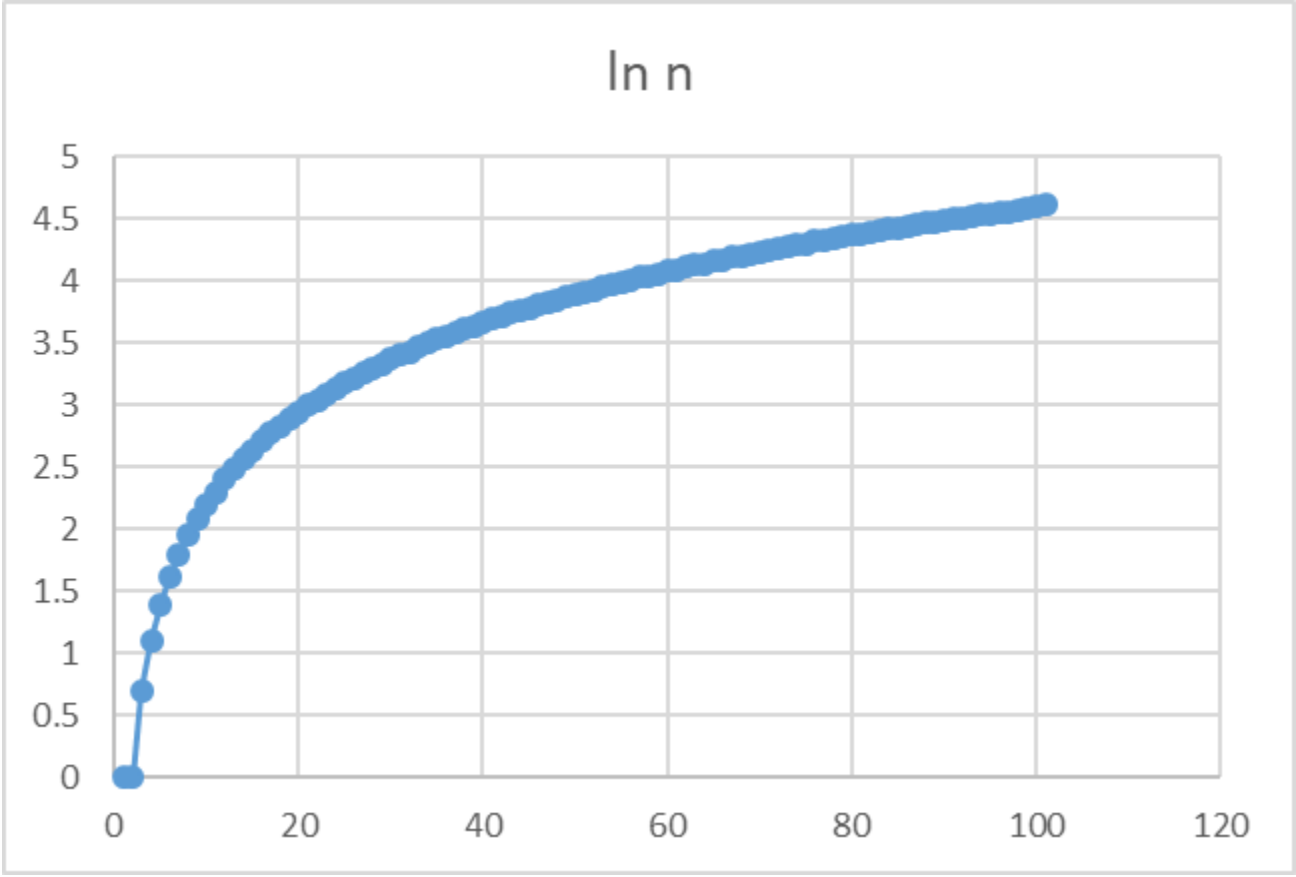
From the above graph we infer that points n=0 and n=1 doesn't exists on the real plane. it could be seen that between the values of n 0-10 , the graph is sparsely clustered and the slope is steep whereas thereafter the graph gets densely clustered and the slope decreases and eventually becomes flatter.



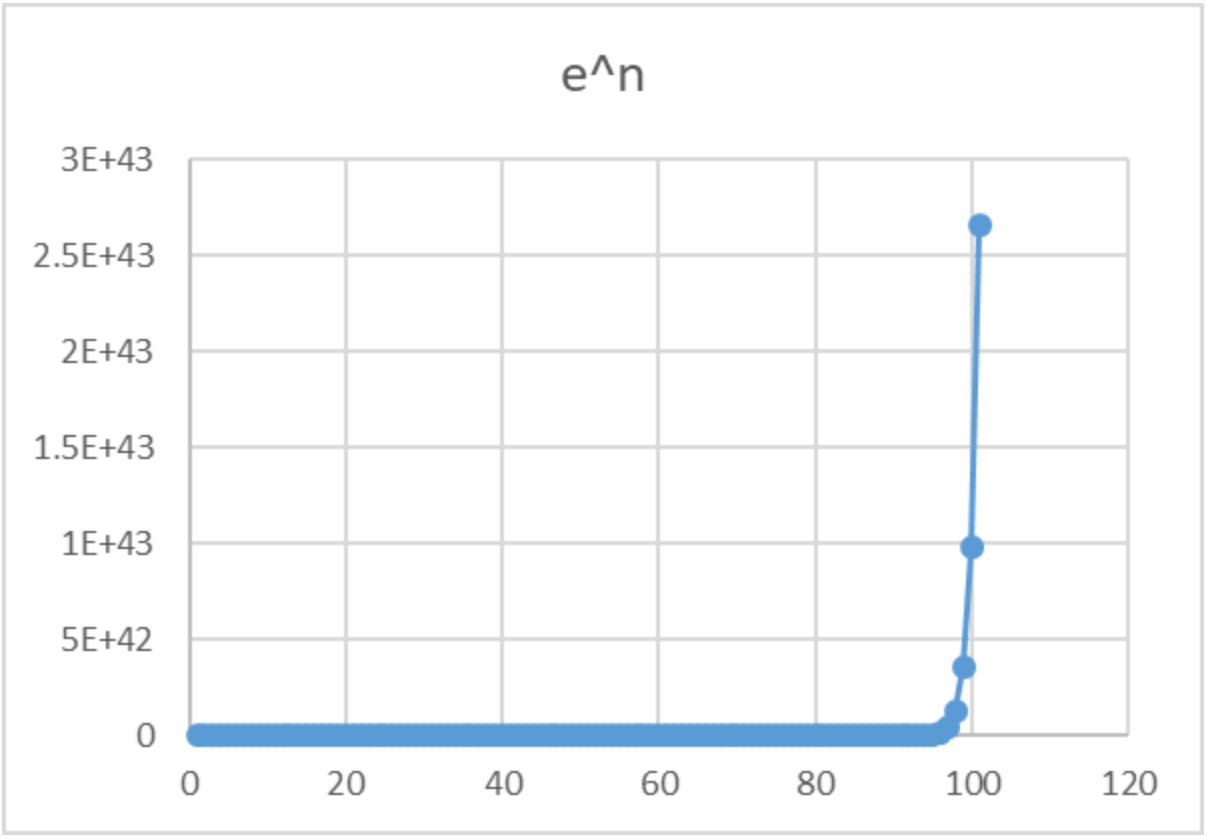
Log 0 doesn’t exists. The graph is rising logarithmically.



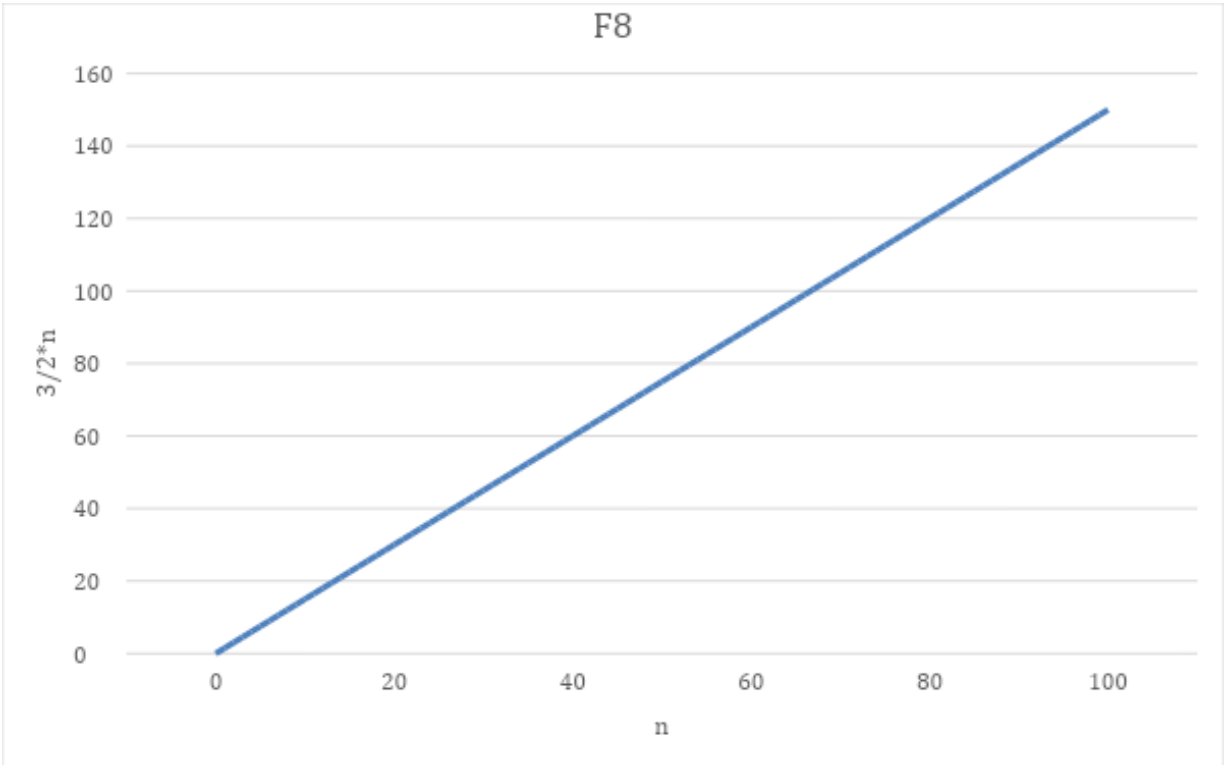
In this graph, we observe that for the values of n between 0 and 20 the graph is less clustered and thereafter it becomes densely clustered with decreasing slope.



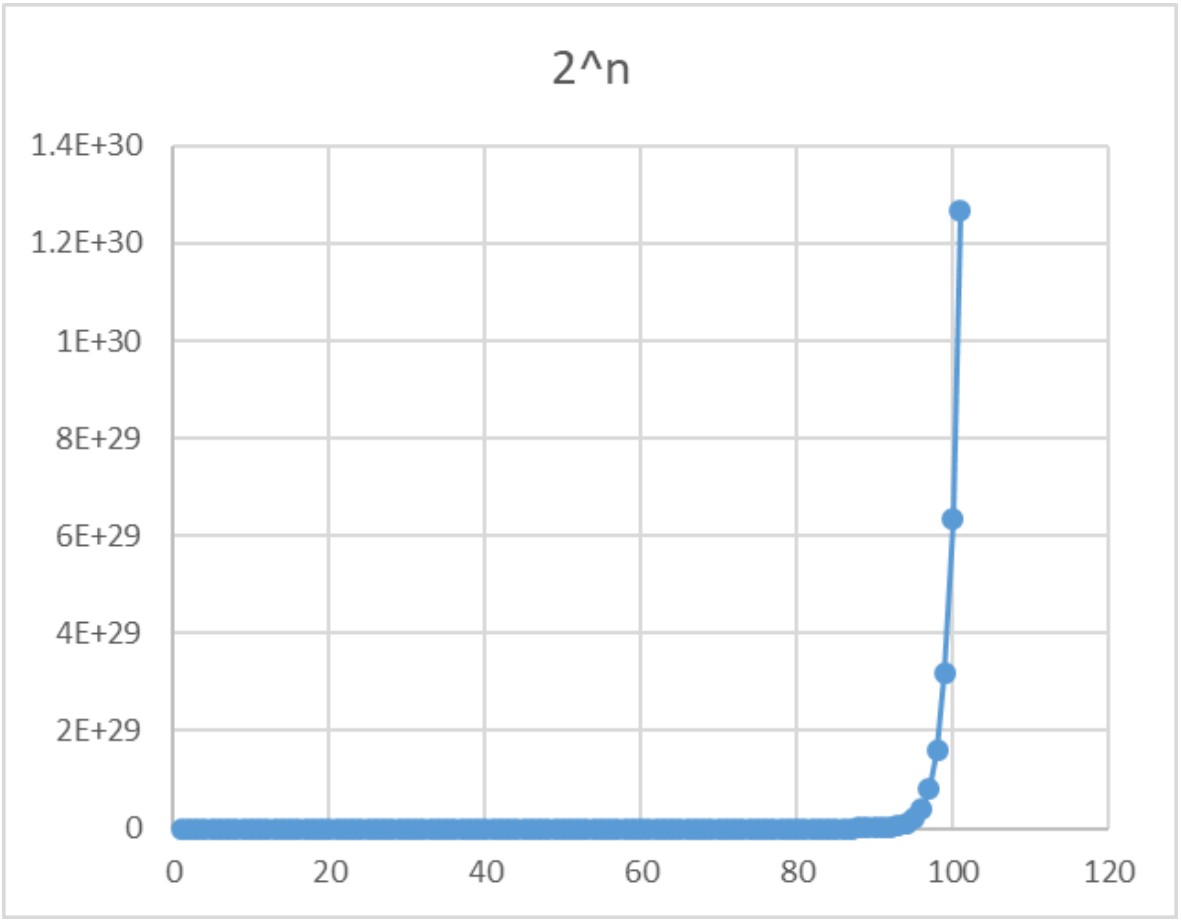
Ln 0 doesn't exists. The graph is rising logarithmically.



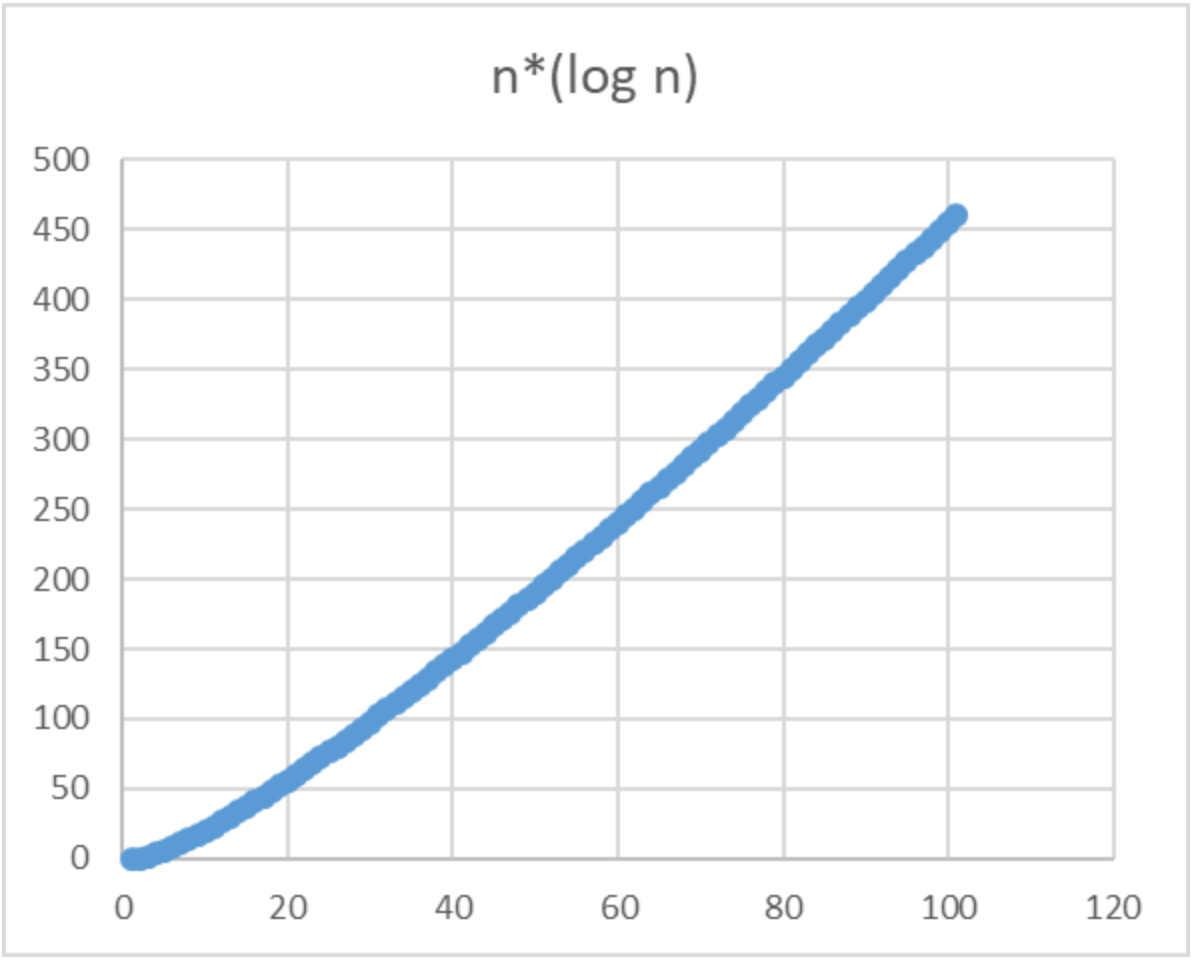
From the above graph we can see that the graph remains almost constant till value of n =95 after which it spikes up drastically and exponentially between 95 and 100.



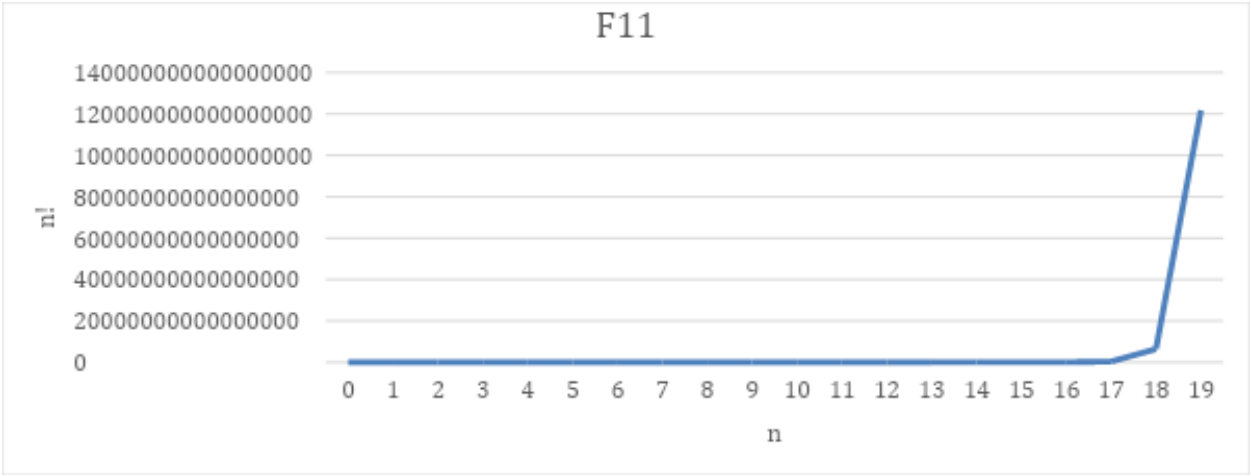
From this graph we can observe that the value of function increases linearly as the value of n increases.



the graph remains almost constant and flat till n= 95 and then it spikes up drastically and exponentially with a almost a 90 degree slope.



The curve in this graph increases somewhat linearly as the value of n increases.



The curve for factorial is linear from 0-17 and then shoots rapidly from 18-19.

CONCLUSION:	In this experiment we have plotted graphs for different functions and

	observed their growth for various cases.