Name	Bhagya Bijlaney
UID no.	2021700010
Experiment No.	1(A)

AIM:	To implement the various functions e.g. linear, non-linear, quadratic, exponential etc.
Program 1	
PROBLEM STATEMENT :	For this experiment, you have to implement at least 10 functions from the following list.
PROGRAM:	

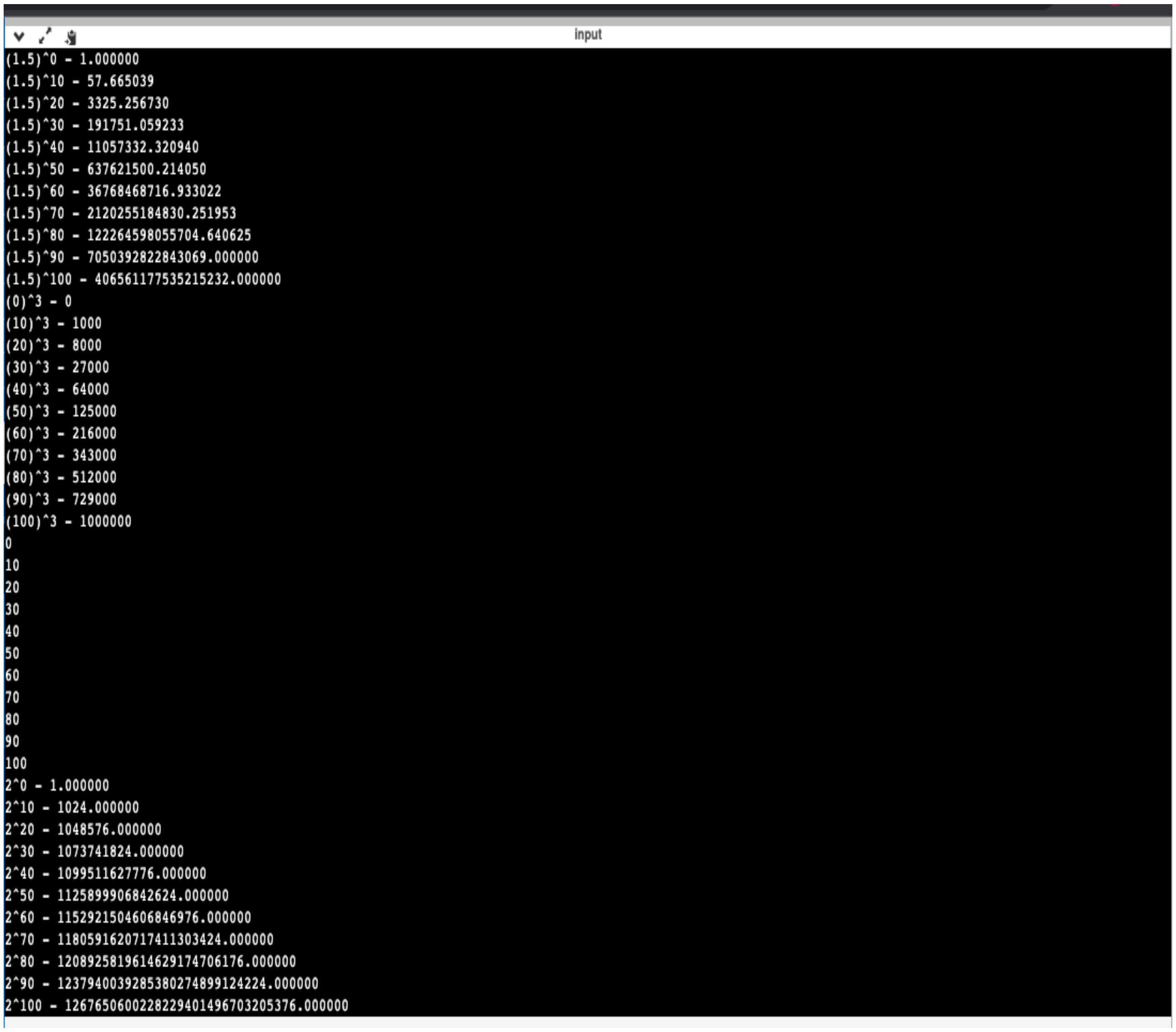
```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

double f1()
{
    double ans;
    for (int n = 0; n <= 100; n = n + 10)
    {
        printf("(1.5)^%d - ", n);
        printf("%f", pow(1.5, n));
        printf("\n");
    }
}
int f2()
{
    int ans;
    for (int n = 0; n <= 100; n = n + 10)
    {
        printf("(%d)^3 - ", n);
        printf("%d", n * n * n);
        printf("\n");
    }
}
int f3()
{
    int ans;
    for (int n = 0; n <= 100; n = n + 10)
    {
        printf("%d ", n);
        printf("\n");
    }
}
double f4()
{
    double ans;
    for (int n = 0; n <= 100; n = n + 10)
    {
        printf("2^%d - ", n);
        printf("%f", pow(2, n));
        printf("\n");
    }
}
double f5()
{
    double ans;
    for (int n = 0; n <= 100; n = n + 10)
    {
        printf("n*(2^%d) - ", n);
        printf("%f", n * pow(2, n));
        printf("\n");
    }
}
double f6()
{
    double ans;
    for (int n = 0; n <= 10; n++)
    {
        printf("2^(2^%d) - ", n);
        printf("%f", pow(2, pow(2, n)));
        printf("\n");
    }
}
double f7()
{
    double ans;
    for (int n = 0; n <= 10; n++)
    {
        printf("log2(%d) - ", n);
        printf("%f", log2(n));
        printf("\n");
    }
```

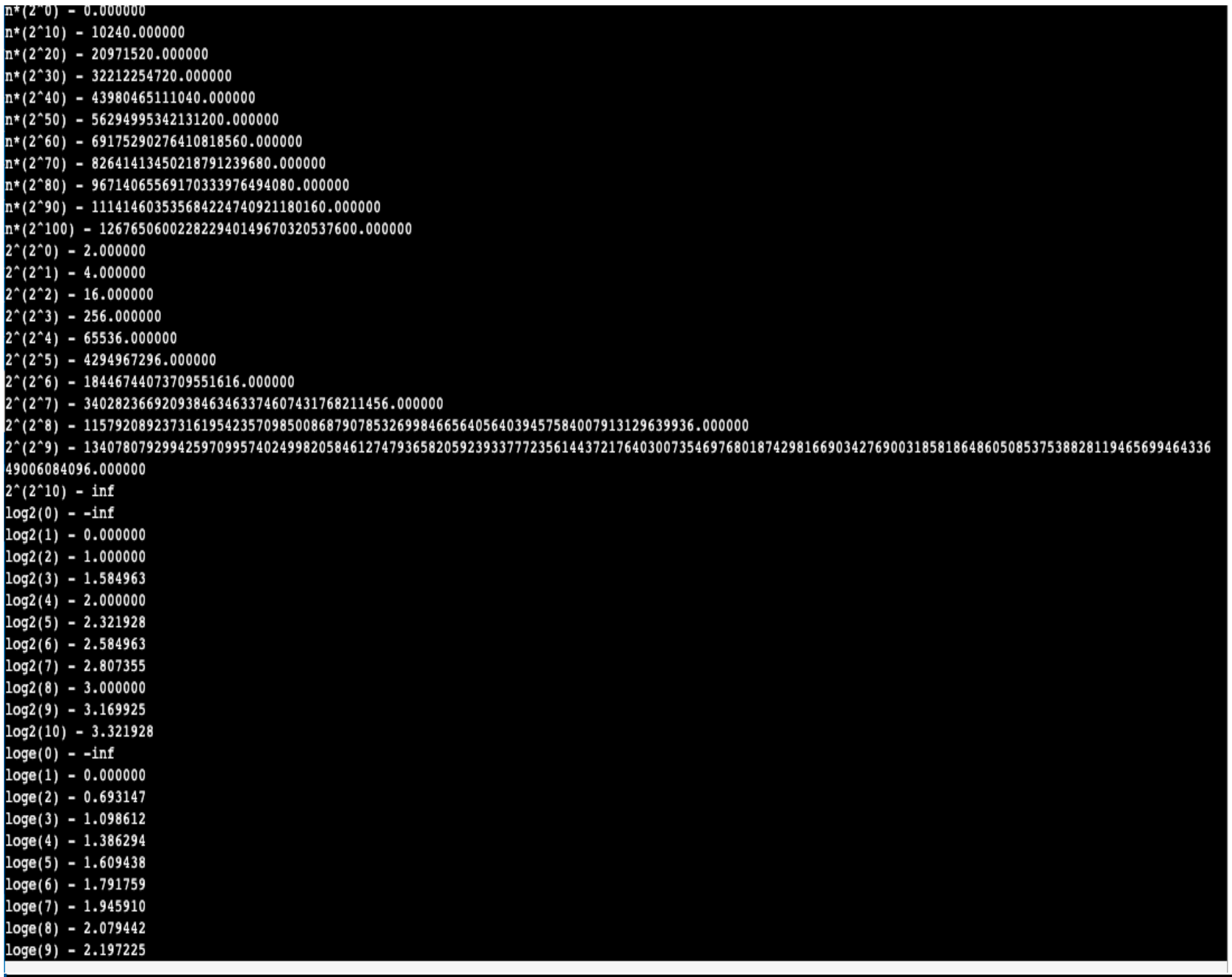
```c
}
double f8()
{
    double ans;
    for (int n = 0; n <= 10; n++)
    {
        printf("loge(%d) - ", n);
        printf("%f", log(n));
        printf("\n");
    }
}
double f9()
{
    double ans;
    for (int n = 0; n <= 10; n++)
    {
        printf("e^(%d) - ", n);
        printf("%f", exp(n));
        printf("\n");
    }
}
double f10()
{
    double ans;
    for (int n = 0; n <= 10; n++)
    {
        printf("n*log2(%d) - ", n);
        printf("%f", n * log2(n));
        printf("\n");
    }
}
int main()
{
    f1();
    f2();
    f3();
    f4();
    f5();
    f6();
    f7();
    f8();
    f9();
    f10();
    return 0;
```

```
}
```

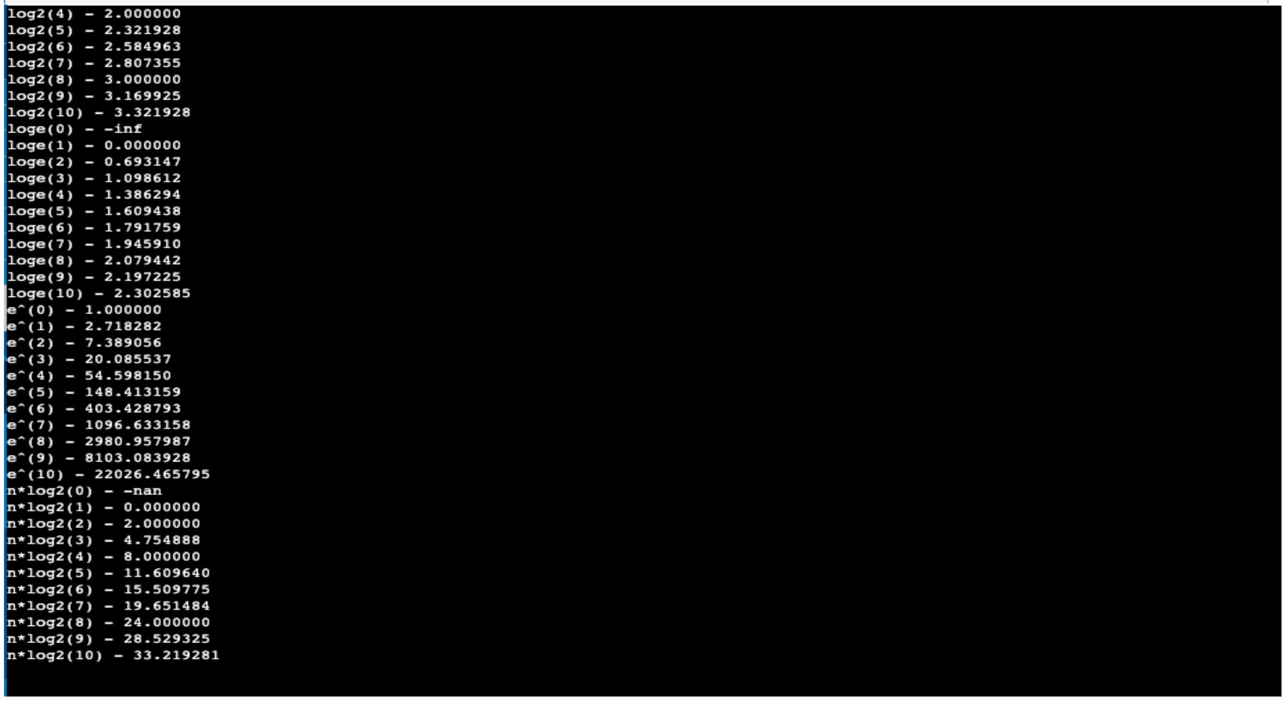
RESULT:



```
(1.5)^0 - 1.000000
(1.5)^10 - 57.665039
(1.5)^20 - 3325.256730
(1.5)^30 - 191751.059233
(1.5)^40 - 11057332.320940
(1.5)^50 - 637621500.214050
(1.5)^60 - 36768468716.933022
(1.5)^70 - 2120255184830.251953
(1.5)^80 - 122264598055704.640625
(1.5)^90 - 7050392822843069.000000
(1.5)^100 - 406561177535215232.000000
(0)^3 - 0
(10)^3 - 1000
(20)^3 - 8000
(30)^3 - 27000
(40)^3 - 64000
(50)^3 - 125000
(60)^3 - 216000
(70)^3 - 343000
(80)^3 - 512000
(90)^3 - 729000
(100)^3 - 1000000
0
10
20
30
40
50
60
70
80
90
100
2^0 - 1.000000
2^10 - 1024.000000
2^20 - 1048576.000000
2^30 - 1073741824.000000
2^40 - 1099511627776.000000
2^50 - 1125899906842624.000000
2^60 - 1152921504606846976.000000
2^70 - 1180591620717411303424.000000
2^80 - 1208925819614629174706176.000000
2^90 - 1237940039285380274899124224.000000
2^100 - 1267650600228229401496703205376.000000
```

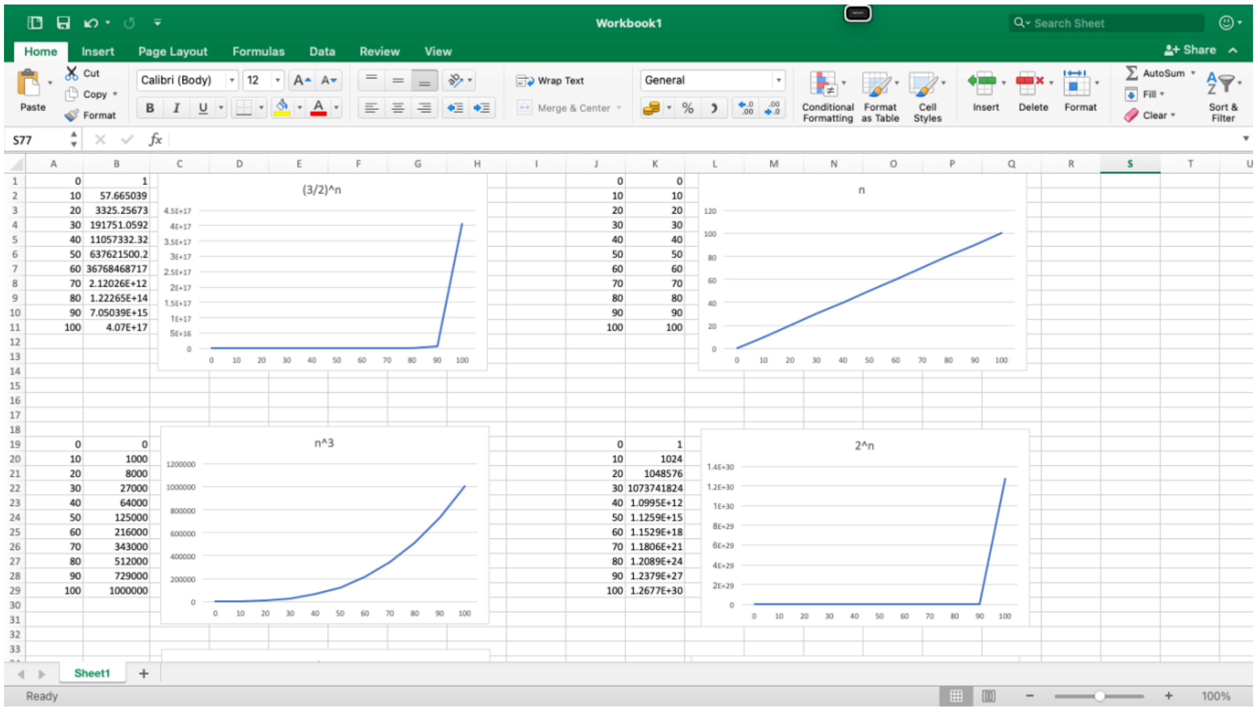


```
n*(2^0) - 0.000000
n*(2^10) - 10240.000000
n*(2^20) - 20971520.000000
n*(2^30) - 32212254720.000000
n*(2^40) - 43980465111040.000000
n*(2^50) - 56294995342131200.000000
n*(2^60) - 69175290276410818560.000000
n*(2^70) - 82641413450218791239680.000000
n*(2^80) - 96714065569170333976494080.000000
n*(2^90) - 111414603535684224740921180160.000000
n*(2^100) - 126765060022822940149670320537600.000000
2^(2^0) - 2.000000
2^(2^1) - 4.000000
2^(2^2) - 16.000000
2^(2^3) - 256.000000
2^(2^4) - 65536.000000
2^(2^5) - 4294967296.000000
2^(2^6) - 18446744073709551616.000000
2^(2^7) - 340282366920938463463374607431768211456.000000
2^(2^8) - 115792089237316195423570985008687907853269984665640564039457584007913129639936.000000
2^(2^9) - 1340780792994259709957402499820584612747936582059239337772356144372176403007354697680187429816690342769003185818648605085375388281194656994643364
9006084096.000000
2^(2^10) - inf
log2(0) - -inf
log2(1) - 0.000000
log2(2) - 1.000000
log2(3) - 1.584963
log2(4) - 2.000000
log2(5) - 2.321928
log2(6) - 2.584963
log2(7) - 2.807355
log2(8) - 3.000000
log2(9) - 3.169925
log2(10) - 3.321928
loge(0) - -inf
loge(1) - 0.000000
loge(2) - 0.693147
loge(3) - 1.098612
loge(4) - 1.386294
loge(5) - 1.609438
loge(6) - 1.791759
loge(7) - 1.945910
loge(8) - 2.079442
loge(9) - 2.197225
```



```
log2(4) - 2.000000
log2(5) - 2.321928
log2(6) - 2.584963
log2(7) - 2.807355
log2(8) - 3.000000
log2(9) - 3.169925
log2(10) - 3.321928
loge(0) - -inf
loge(1) - 0.000000
loge(2) - 0.693147
loge(3) - 1.098612
loge(4) - 1.386294
loge(5) - 1.609438
loge(6) - 1.791759
loge(7) - 1.945910
loge(8) - 2.079442
loge(9) - 2.197225
```

```
loge(10) - 2.302585
e^(0) - 1.000000
e^(1) - 2.718282
e^(2) - 7.389056
e^(3) - 20.085537
e^(4) - 54.598150
e^(5) - 148.413159
e^(6) - 403.428793
e^(7) - 1096.633158
e^(8) - 2980.957987
e^(9) - 8103.083928
e^(10) - 22026.465795
n*log2(0) - -nan
n*log2(1) - 0.000000
n*log2(2) - 2.000000
n*log2(3) - 4.754888
n*log2(4) - 8.000000
n*log2(5) - 11.609640
n*log2(6) - 15.509775
n*log2(7) - 19.651484
n*log2(8) - 24.000000
n*log2(9) - 28.529325
n*log2(10) - 33.219281
```

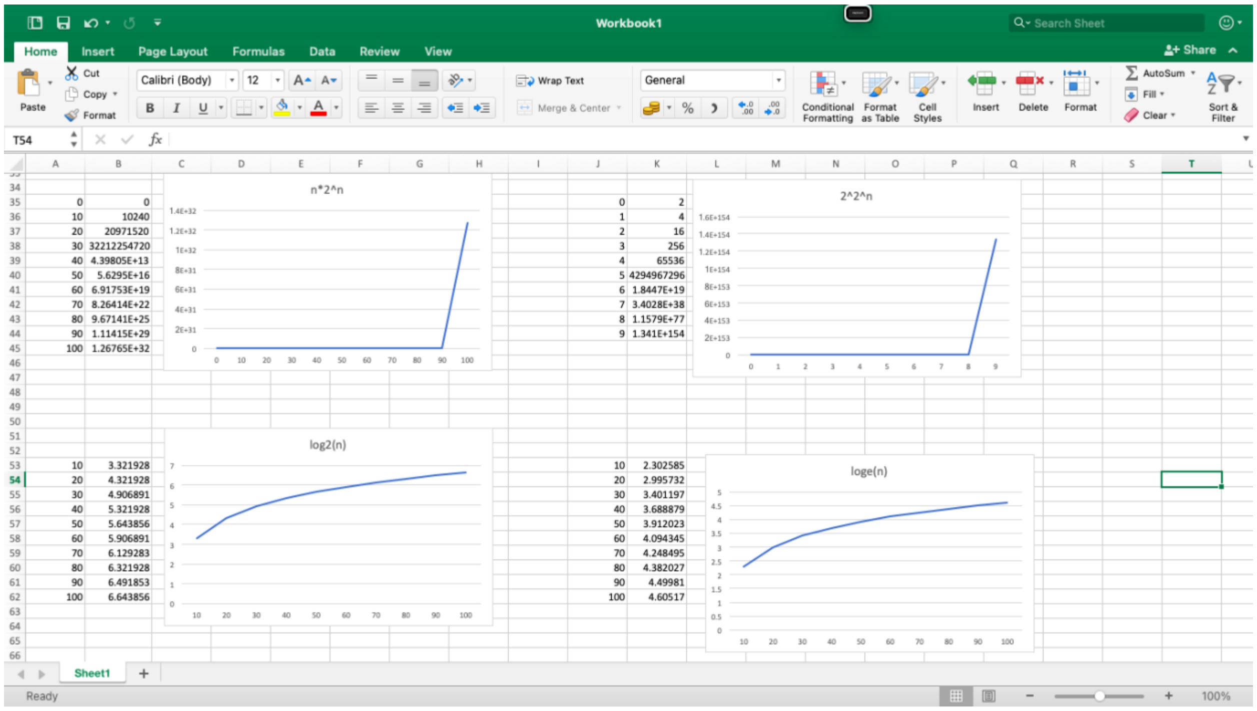


	A	B
1	0	1
2	10	57.665039
3	20	3325.25673
4	30	191751.0592
5	40	11057332.32
6	50	637621500.2
7	60	36768468717
8	70	2.12026E+12
9	80	1.22265E+14
10	90	7.05039E+15
11	100	4.07E+17

	J	K
1	0	0
2	10	10
3	20	20
4	30	30
5	40	40
6	50	50
7	60	60
8	70	70
9	80	80
10	90	90
11	100	100

	A	B
19	0	0
20	10	1000
21	20	8000
22	30	27000
23	40	64000
24	50	125000
25	60	216000
26	70	343000
27	80	512000
28	90	729000
29	100	1000000

	J	K
19	0	1
20	10	1024
21	20	1048576
22	30	1073741824
23	40	1.0995E+12
24	50	1.1259E+15
25	60	1.1529E+18
26	70	1.1806E+21
27	80	1.2089E+24
28	90	1.2379E+27
29	100	1.2677E+30

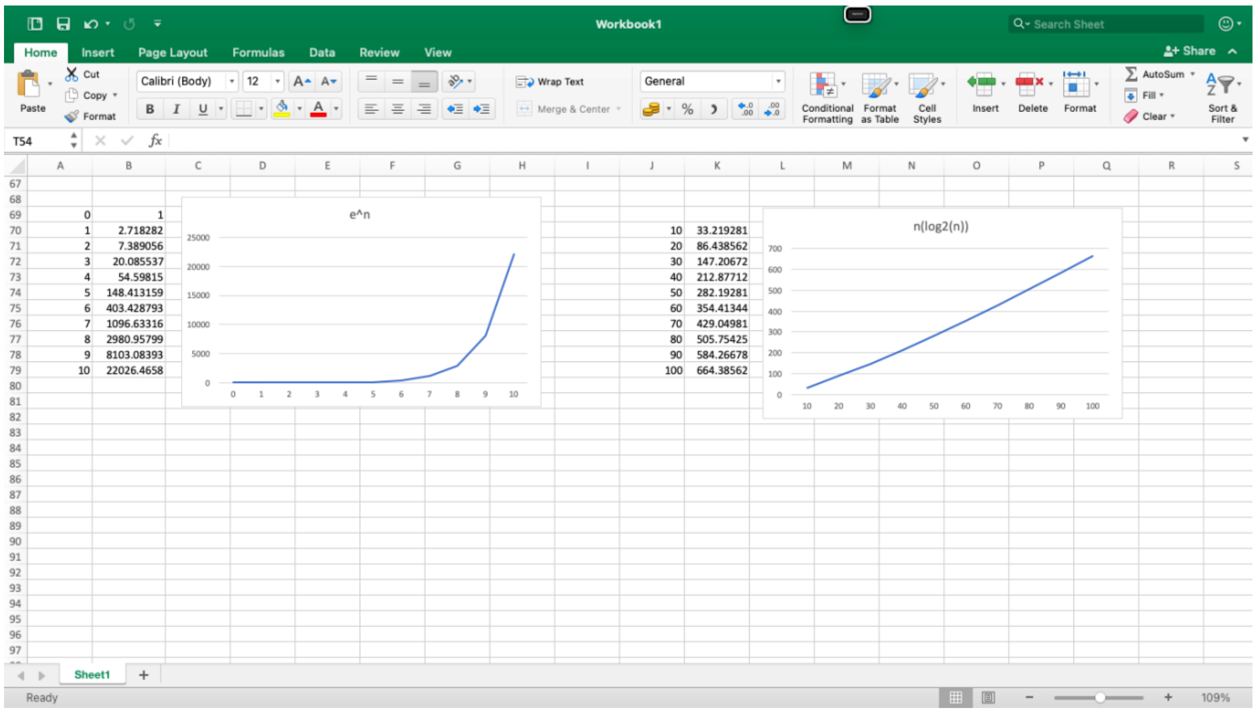


	A	B
35	0	0
36	10	10240
37	20	20971520
38	30	32212254720
39	40	4.39805E+13
40	50	5.6295E+16
41	60	6.91753E+19
42	70	8.26414E+22
43	80	9.67141E+25
44	90	1.11415E+29
45	100	1.26765E+32

	J	K
35	0	2
36	1	4
37	2	16
38	3	256
39	4	65536
40	5	4294967296
41	6	1.8447E+19
42	7	3.4028E+38
43	8	1.1579E+77
44	9	1.341E+154

	A	B
53	10	3.321928
54	20	4.321928
55	30	4.906891
56	40	5.321928
57	50	5.643856
58	60	5.906891
59	70	6.129283
60	80	6.321928
61	90	6.491853
62	100	6.643856

	J	K
53	10	2.302585
54	20	2.995732
55	30	3.401197
56	40	3.688879
57	50	3.912023
58	60	4.094345
59	70	4.248495
60	80	4.382027
61	90	4.49981
62	100	4.60517



0	1
1	2.718282
2	7.389056
3	20.085537
4	54.59815
5	148.413159
6	403.428793
7	1096.63316
8	2980.95799
9	8103.08393
10	22026.4658

10	33.219281
20	86.438562
30	147.20672
40	212.87712
50	282.19281
60	354.41344
70	429.04981
80	505.75425
90	584.26678
100	664.38562

CONCLUSION:	We understood the basics of functions and graphs in excel.