DAA Experiment-1-A

(Batch-A/A1)

Name	Ansari Mohammed Shanouf Valijan
UID Number	2021300004
Class	SY B.Tech Computer Engineering(Div-A)
Experiment Number	1-A

Aim:

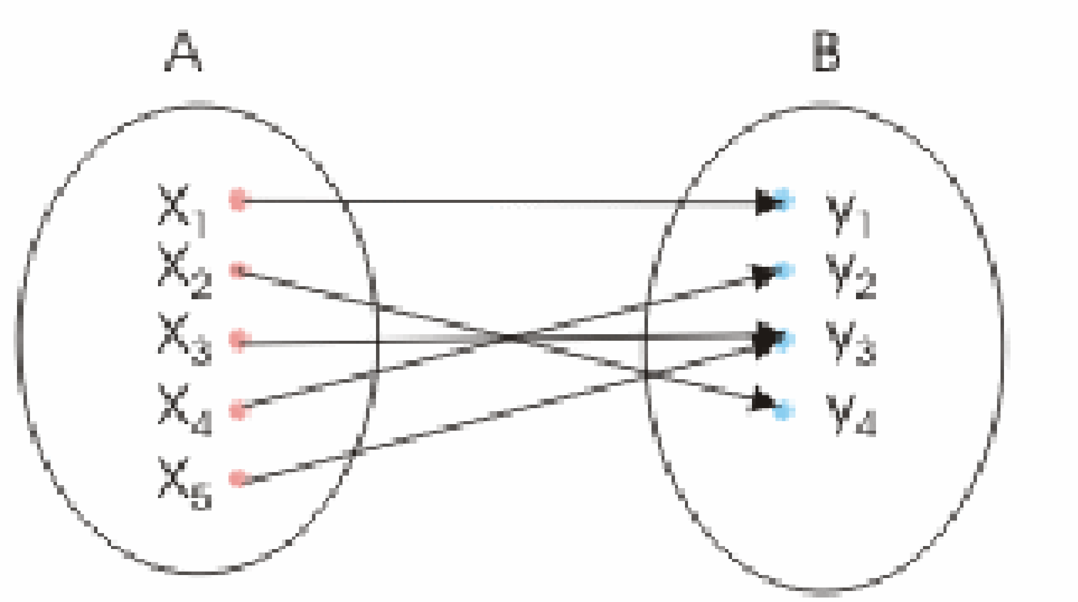
To implement the various linear and non-linear functions.

Problem Definition and Assumptions:

From the given list of functions, implement at-least 10, print the input-output table and plot the graphs. Write your observations.

Theory:

A function in the context of this experiment is a mathematical expression that gives certain outputs based on the inputs provided. A proper understanding of functions in general and their nature would help us while dealing with the time complexities of algorithms. It would help us to compare two or more algorithms in the process of determining the most efficient one.



The image above shows a conceptual view of a function. Basically, inputs are mapped to outputs in a function such that one input will have only one output. However, an output may be generated by two or more inputs.

Algorithms:

[A] For log functions-

I. Call the log function from the 'math.h' header file.
II. Wherever log to the base x is asked, divide the answer obtained in the previous step by log(x) and return the obtained value.

[B] For other functions-

I. Simply call the required function from the math header file and return the obtained value.

[C] For factorial function-

I. Return the provided number multiplied by factorial of a number one less than the given number till 0 is reached where 1 is returned.

Program:

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

//utility functions
double factorial(double n){
    if(n<=1)
        return 1;
    return n*factorial(n-1);
}

double func1(double x){
    return pow(1.5,x);
}

double func2(double x){
    return pow(x,3);
}

double func3(double x){
    return log(log(x));
}

double func4(double x){
    return log(factorial(x))/log(2);
}

double func5(double x){
    return exp(x);
}

double func6(double x){
```

```
    return log(log(x)/log(2))/log(2);
}

double func7(double x){
    return x;
}

double func8(double x){
    return pow(2,pow(2,x));
}

double func9(double x){
    return x*pow(2,x);
}

double func10(double x){
    return log(x)/log(2);
}

double func11(double x){
    return factorial(x);
}

//main function
void main(){
    for(double i=0; i<=100; i++){
        printf("%.0lf\t%.2lf\n",i,func1(i));
    }
}
```

Note that in the main() function, all the 11 functions(including the factorial function) were executed one after the other and their outputs were copied in an excel file for obtaining their respective graphs.

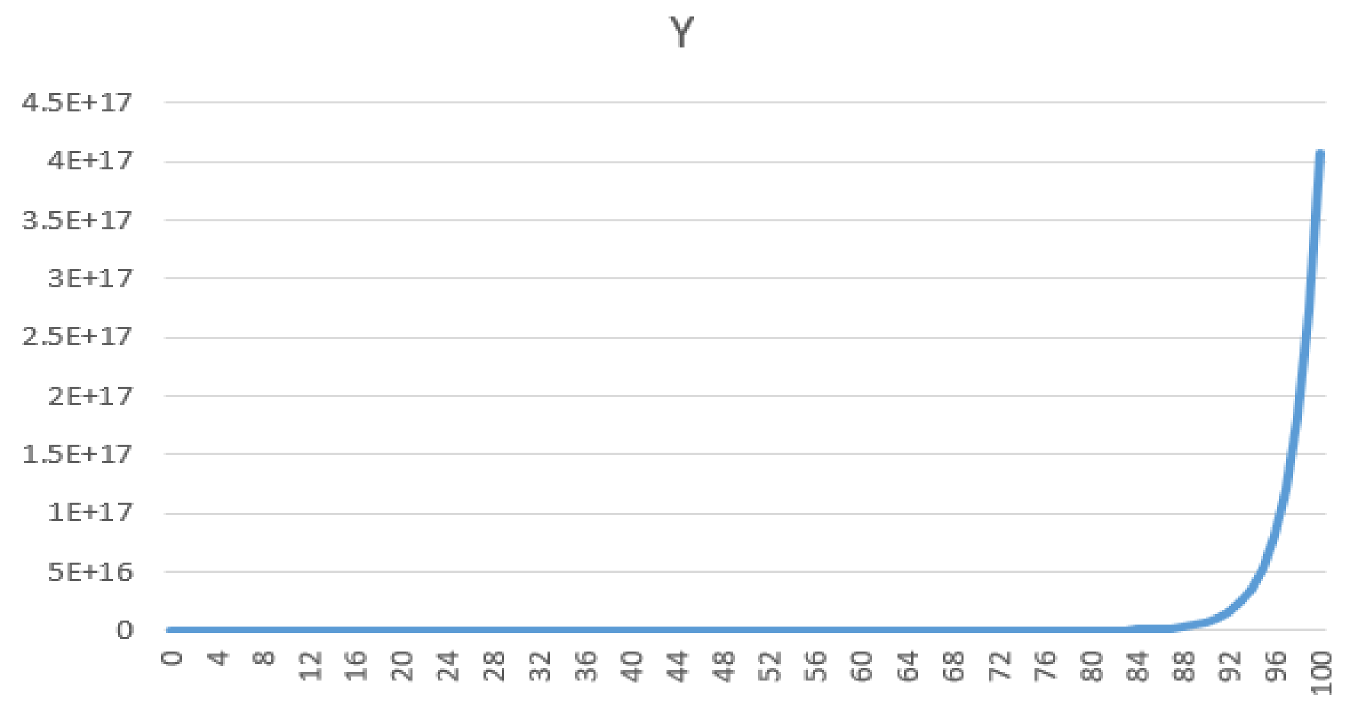
Implementation:

From the given list of functions, the following ten functions were chosen for execution. The corresponding graph obtained and observation for each of the chosen functions are included as well.

The graphs were obtained through the excel file, the screenshots of which are included at the end of the document.

All the x-y values of the functions were pasted from the terminal to the file in order to be able to obtain their respective graphs. In total, eleven functions were executed.

[A] Function-1: $\left(\frac{3}{2}\right)^n$

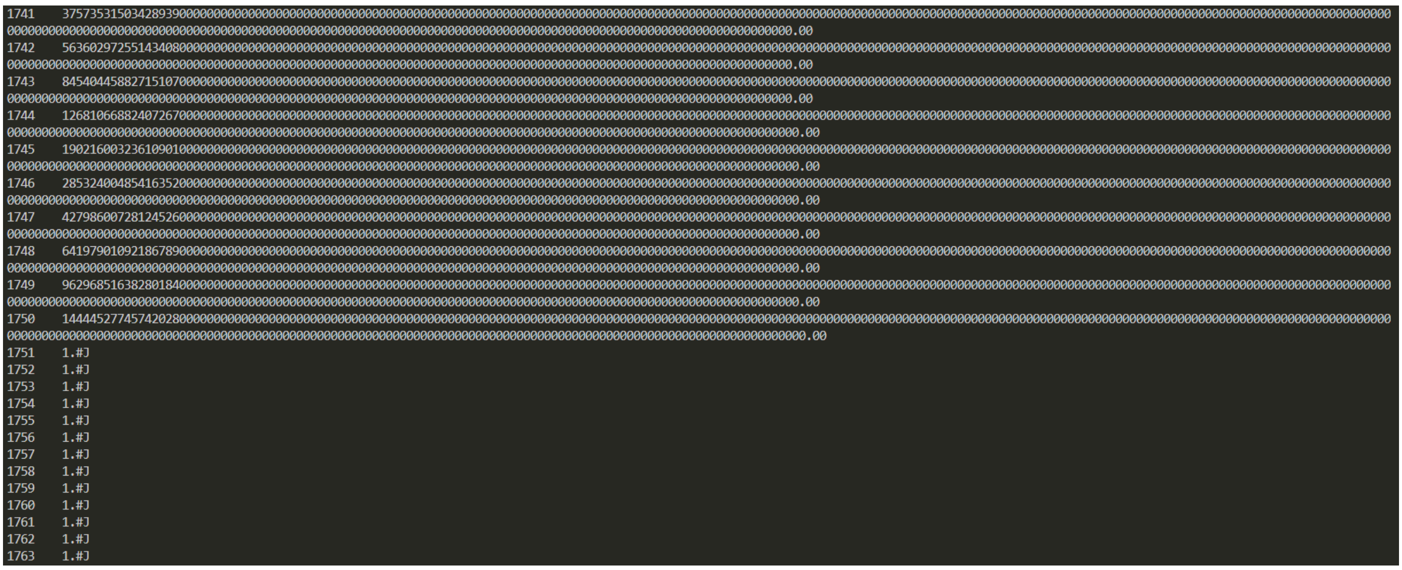


<u>Observations</u>-

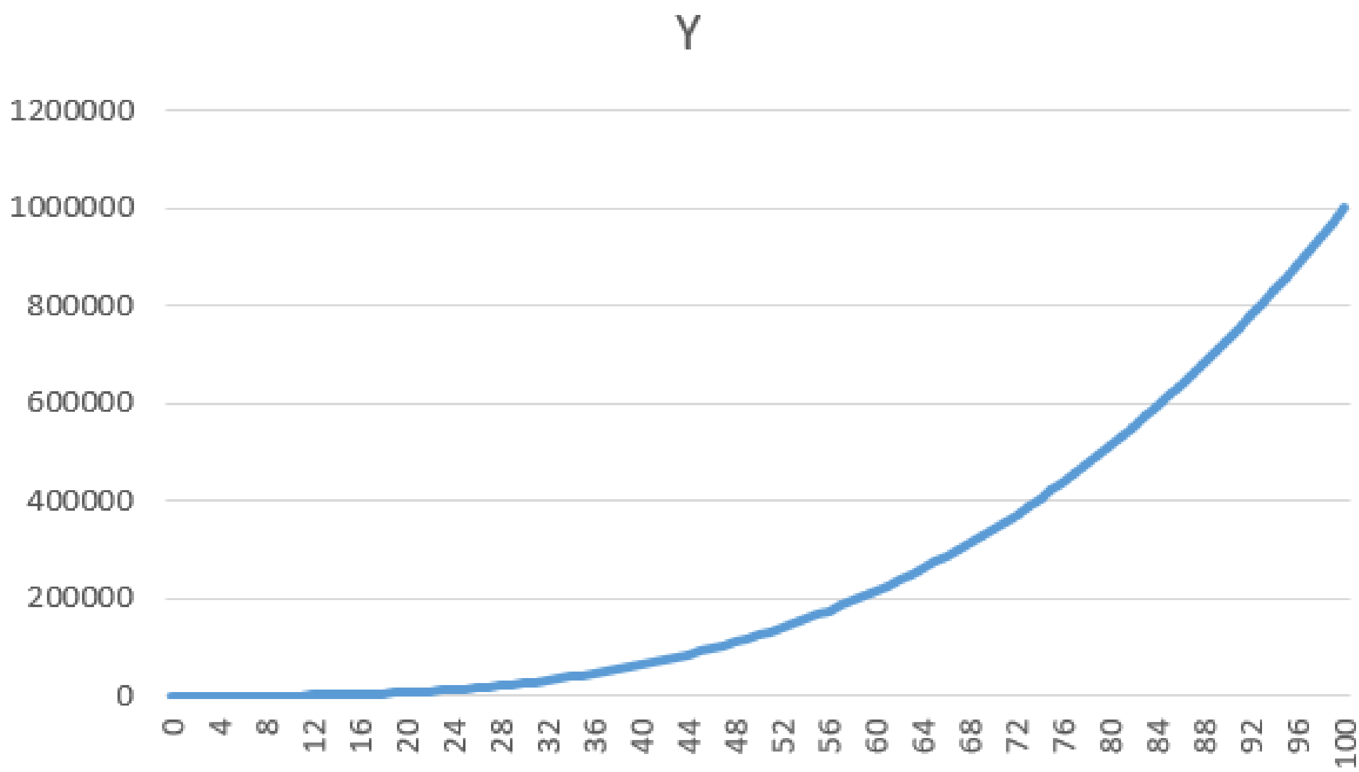
I. The graph seems to have a sudden steep rise thereby indicating that we are dealing with an exponential function.

II. During the runtime of the function on the terminal, it took comparatively more and more time to get the corresponding y value for an x value.

III. By executing the function for values beyond 100, it was observed that a proper output was obtained for values of x up to 1750 after which vague values were obtained. The following image shows the output obtained when the function was executed for values beyond 1750.



[B] Function-2: n^3

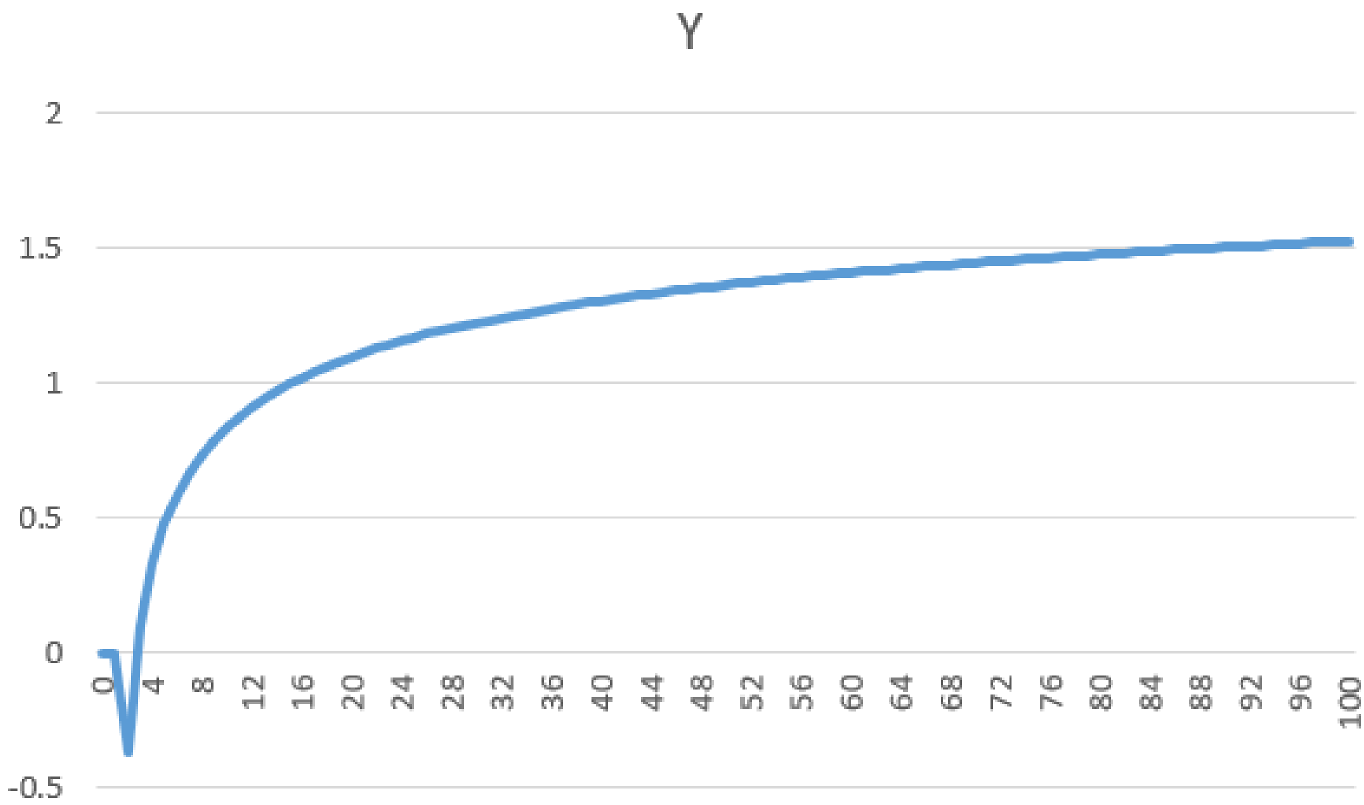


<u>Observations</u>-

I. The graph of the function smoothly increases without any sudden rise or fall as seen above.
II. The curve is concaving in the upwards direction.
III. The function, upon its execution with values much larger than 100, still provides a proper output within the split of a second. The image below shows the output obtained when the function is executed for values in the range of a million.

```
1000963 1002891783000056300.00
1000964 1002894788783841300.00
1000965 1002897794573632100.00
1000966 1002900800369428700.00
1000967 1002903806171231100.00
1000968 1002906811979039200.00
1000969 1002909817792853200.00
1000970 1002912823612673000.00
1000984 1002954905720763900.00
1000985 1002957911630671600.00
1000986 1002960917546585200.00
1000987 1002963923468504800.00
1000988 1002966929396430300.00
1000989 1002969935330361700.00
1000990 1002972941270299000.00
1000991 1002975947216242300.00
1000992 1002978953168191500.00
1000993 1002981959126146700.00
1000994 1002984965090107800.00
1000995 1002987971060074900.00
1000996 1002990977036047900.00
1000997 1002993983018027000.00
1000998 1002996989006012000.00
1000999 1002999995000002900.00
1001000 1003003001000000000.00
```

[C] Function-3: $\ln(\ln n)$



<u>Observations-</u>

I. This function is not defined for the input values 0 and 1 as can be seen in the excel screenshot at the end. For these inputs, the output is treated as default by the excel and hence the above graph starts as a flat line.

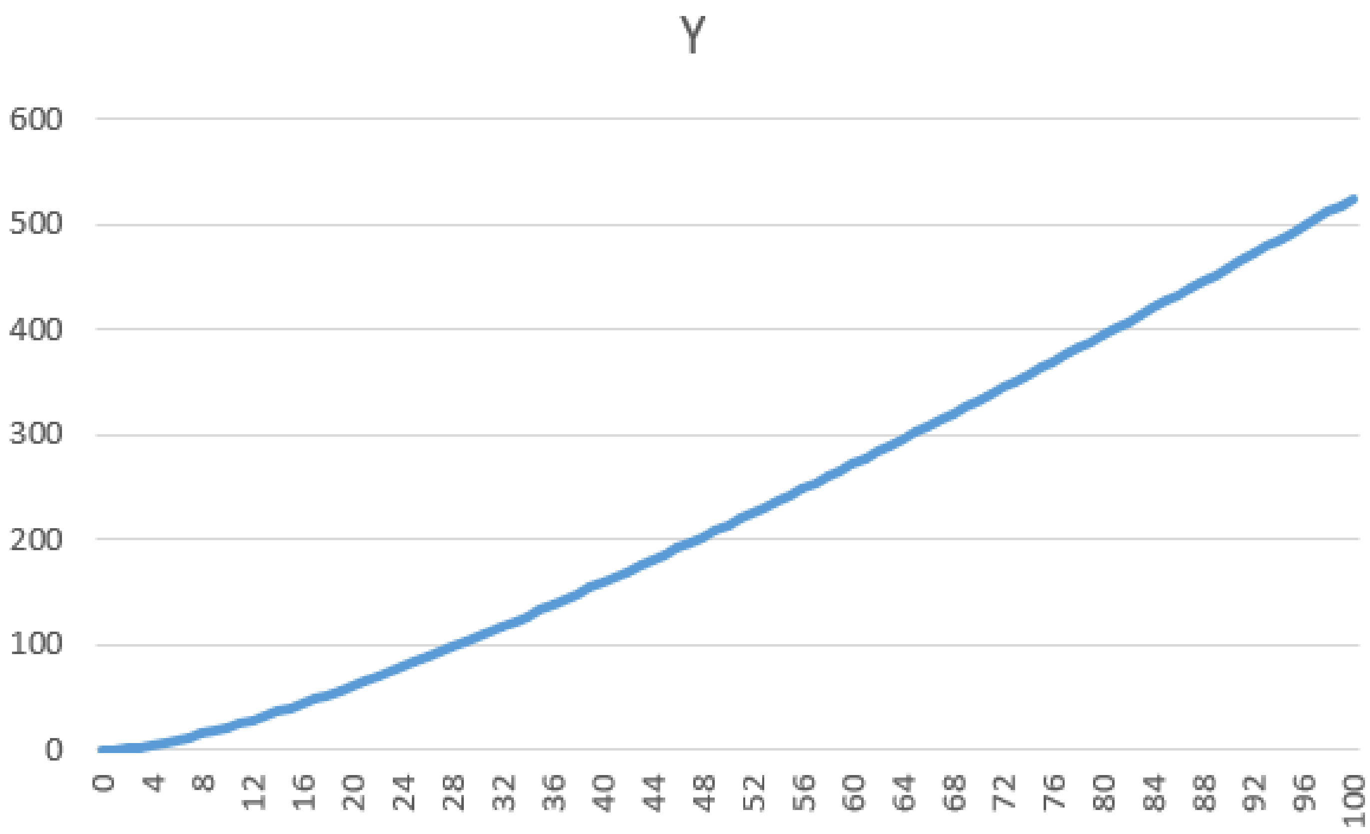
II. The graph then steeps down to the only negative value output in the range from 0 to 100.

III. Further, the graph quickly rises up, then gradually slows down, thereby depicting the behaviour of a logarithmic function.

IV. From the execution of the graph, it is found that it limits to the approximate value of 2.63 at large inputs that are in the range of a million.

```
1000170 2.63
1000171 2.63
1000172 2.63
1000173 2.63
1000174 2.63
1000175 2.63
1000176 2.63
1000177 2.63
1000178 2.63
1000179 2.63
1000180 2.63
1000181 2.63
1000182 2.63
1000183 2.63
1000184 2.63
1000185 2.63
1000186 2.63
1000187 2.63
1000188 2.63
```

[D] Function-4: $\log_2(n!)$



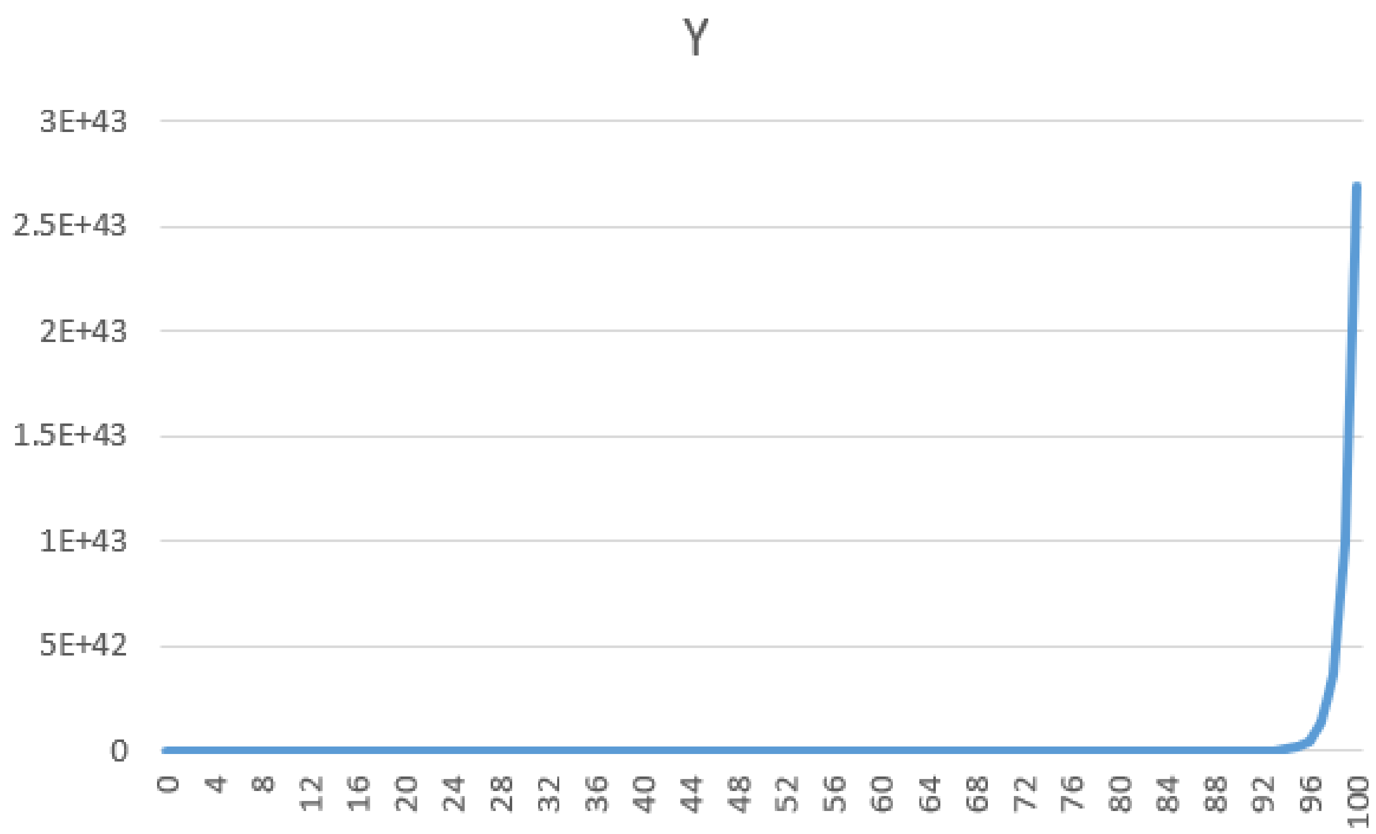
Observations-

I. An almost linear behaviour of the function is observed through its graph plotted for the inputs from 0 to 100.

II. Due to the primary part of the function being factorial, we observe that vague values are obtained once the input crosses the value of 170. This is shown in the image below.

```
166     989.78
167     997.17
168     1004.56
169     1011.96
170     1019.37
171     1.#J
172     1.#J
173     1.#J
174     1.#J
175     1.#J
176     1.#J
177     1.#J
178     1.#J
179     1.#J
180     1.#J
181     1.#J
182     1.#J
183     1.#J
184     1.#J
185     1.#J
186     1.#J
187     1.#J
188     1.#J
```

[E] Function-5: e^n



<u>Observations</u>-

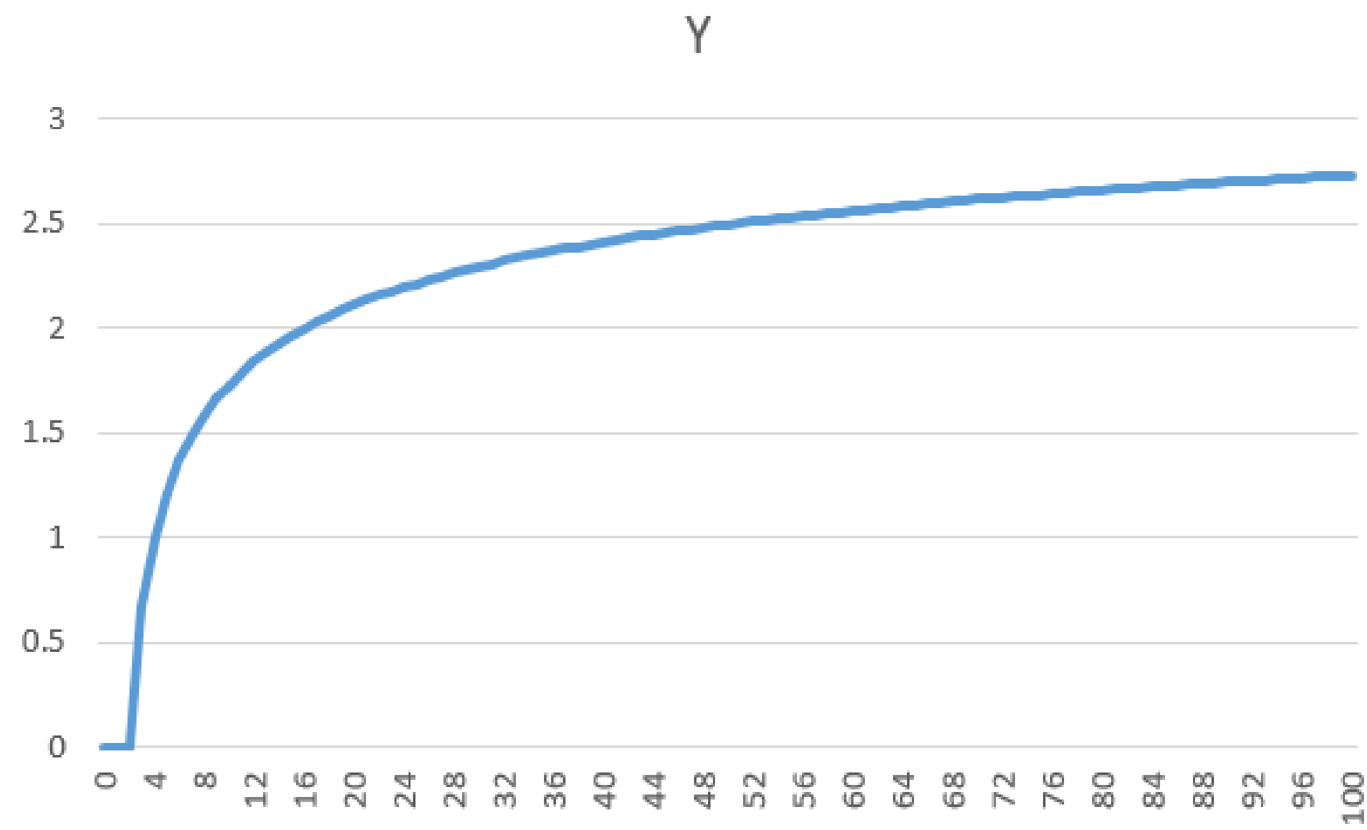
I. In the plot of the graph for input ranging from 0 to 100, a flat line is observed which suddenly rises around the input value of 96, showing a very steep behaviour.

II. The sudden shift of output demonstrates the nature of an exponential function.

III. Upon executing the function for inputs beyond the value of 100, it is found that we start getting vague outputs after the value of 709. This is shown below.

```
704     55375193892845935000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
705     150525383306319410000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
706     409170414163400540000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
707     1112240501563433300000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
708     3023383144276055200000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
709     8218407461554972400000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
710     1.#J
711     1.#J
712     1.#J
713     1.#J
714     1.#J
715     1.#J
716     1.#J
717     1.#J
718     1.#J
719     1.#J
720     1.#J
721     1.#J
722     1.#J
723     1.#J
724     1.#J
725     1.#J
726     1.#J
```

[F] Function-6: $\log_2(\log_2 n)$

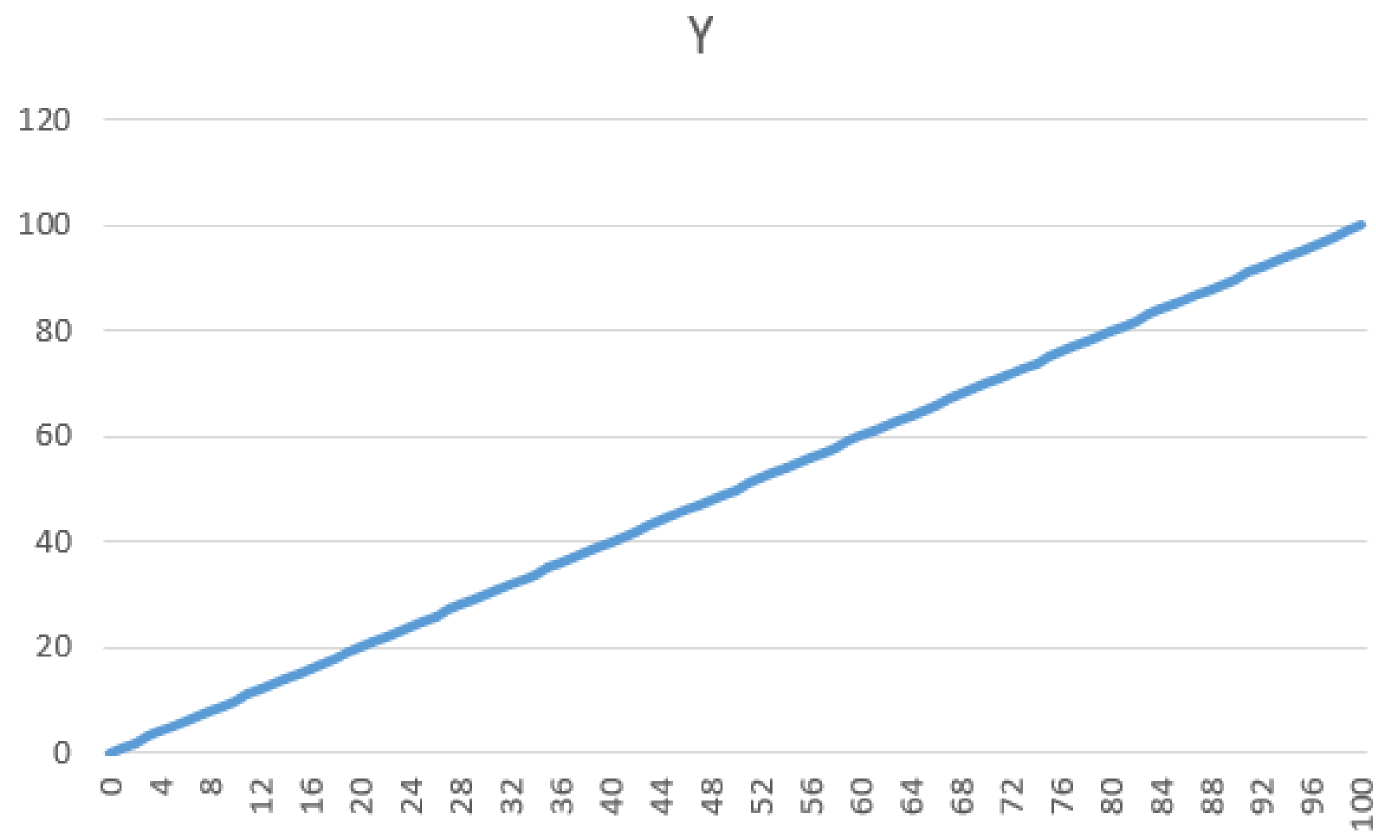


<u>Observations</u>-

I. This particular function is undefined for the input values 0 and 1, the output of which are shown as zero in the graph(the default output for excel).
II. Further, the graph increases, first rapidly, then slowly, concaving in the downward direction.
III. At the input values that fall in the range of a million, we find that the output limits to a value of 4.32. This is illustrated below.

```
1000931 4.32
1000932 4.32
1000933 4.32
1000934 4.32
1000935 4.32
1000936 4.32
1000937 4.32
1000938 4.32
1000939 4.32
1000940 4.32
1000941 4.32
1000942 4.32
1000943 4.32
1000944 4.32
1000945 4.32
1000946 4.32
1000947 4.32
1000948 4.32
1000949 4.32
1000950 4.32
1000951 4.32
1000952 4.32
```

[G] Function-7: n



Observations-

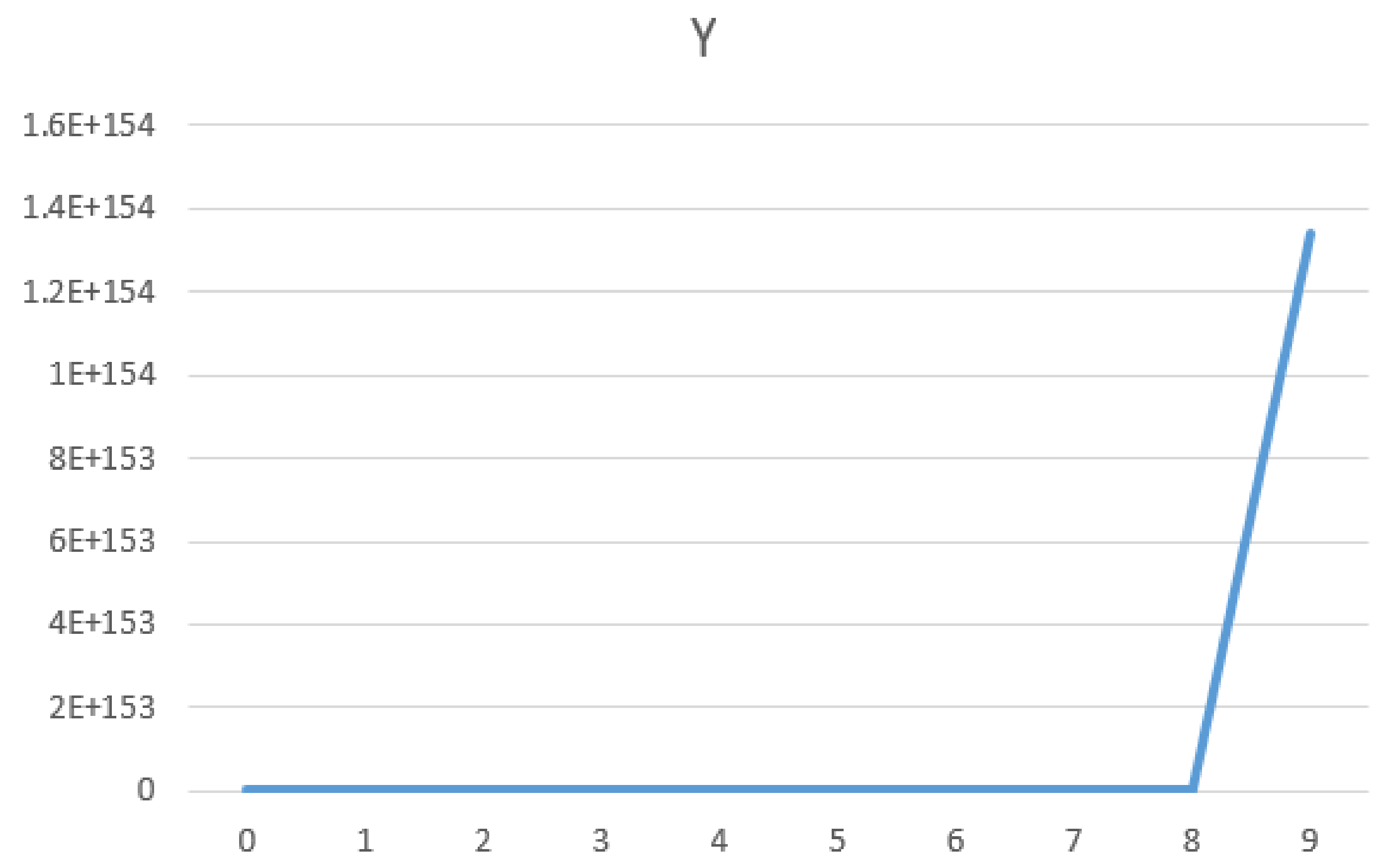
I. A straight line is obtained as we are dealing with a linear function in this case.

II. Upon the execution of this function for input values that are much larger than 100, the outputs are quickly obtained as the function is linear.

III. There are no input values for which the output is undefined. Given below is a sample output of the execution of this function.

```
1542939 1542939.00
1542940 1542940.00
1542941 1542941.00
1542942 1542942.00
1542943 1542943.00
1542944 1542944.00
1542945 1542945.00
1542946 1542946.00
1542947 1542947.00
1542948 1542948.00
1542949 1542949.00
1542950 1542950.00
1542951 1542951.00
1542952 1542952.00
1542953 1542953.00
1542954 1542954.00
1542955 1542955.00
1542956 1542956.00
```

[H] Function-8: 2^{2^n}



Observations-

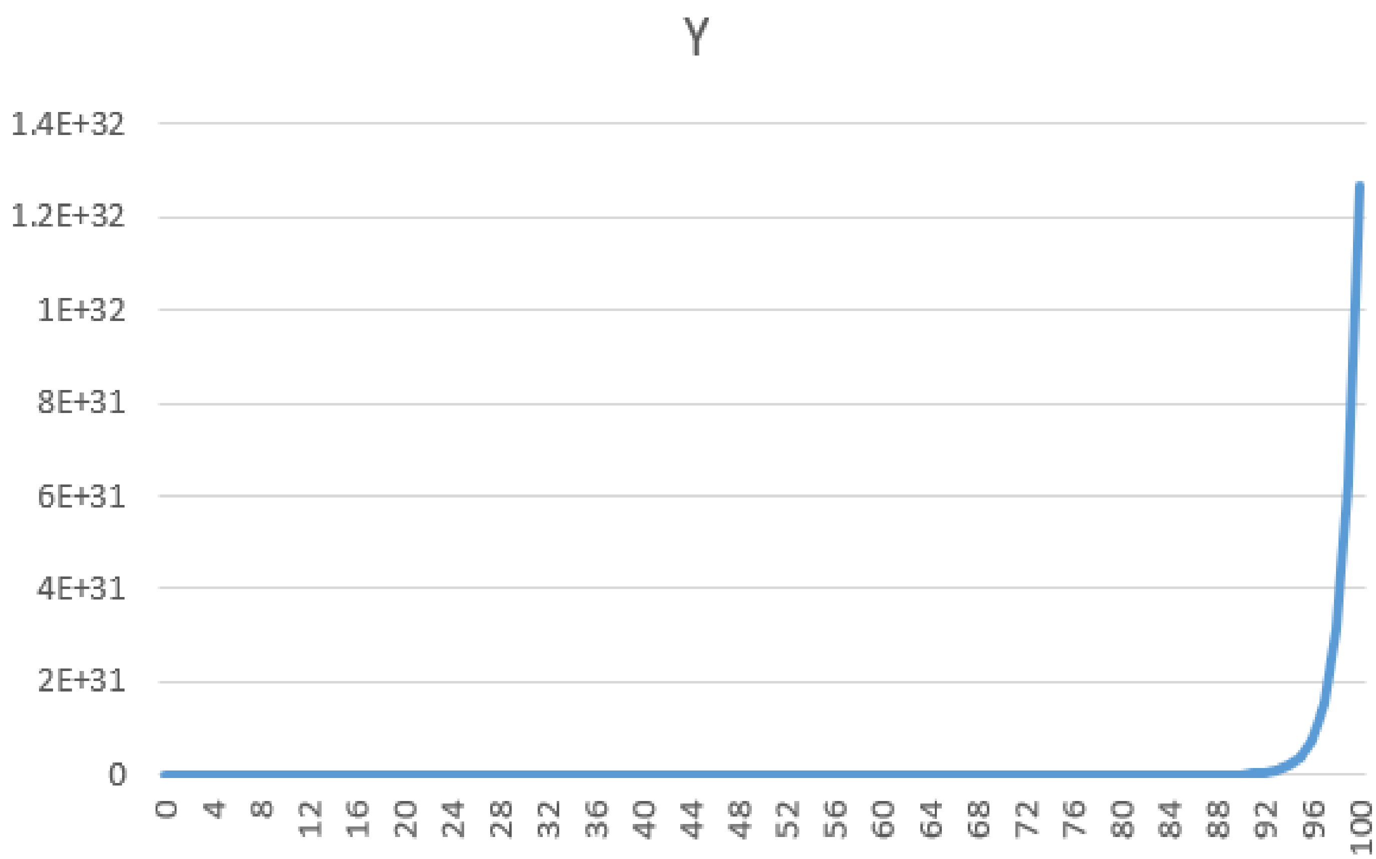
I. The above graph demonstrates that this function increases very rapidly even when the increment in the input is very less.

II. From the execution of this function on the terminal, it was noticed that for input values greater than 9, vague outputs were obtained. This is shown in the following image.

III. From the graph, we also see a sharp increase of the output when the input changes from 8 to 9.

```
0       2.00
1       4.00
2       16.00
3       256.00
4       65536.00
5       4294967296.00
6       18446744073709552000.00
7       340282366920938460000000000000000000000.00
8       115792089237316200000000000000000000000000000000000000000000000000000000000000.00
9       13407807929942597000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
10      1.#J
11      1.#J
12      1.#J
13      1.#J
14      1.#J
15      1.#J
16      1.#J
17      1.#J
18      1.#J
19      1.#J
20      1.#J
```

[I] Function-9: n.2^{n}



<u>Observations</u>-

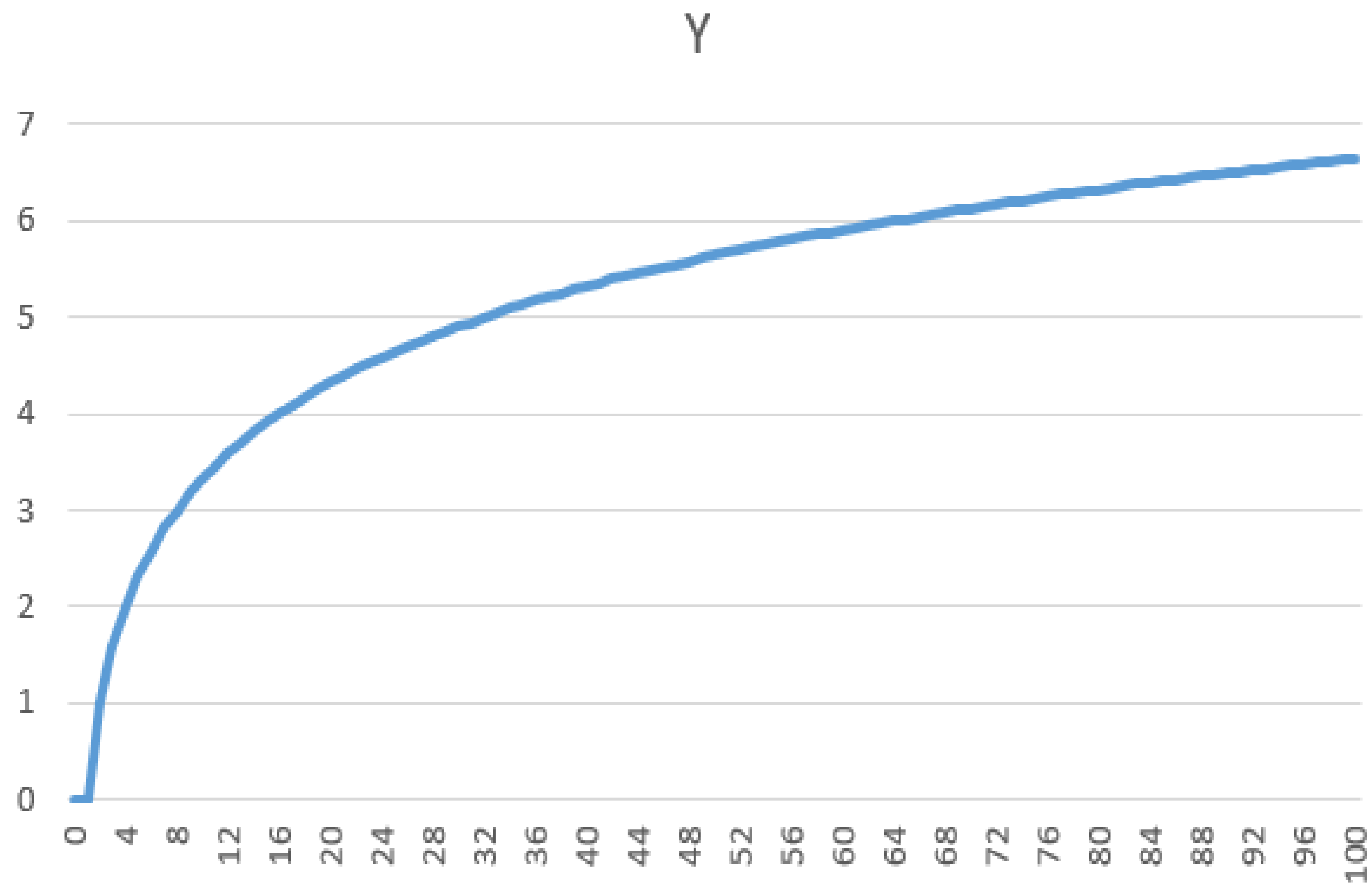
I. The above graph is noted to be similar to that of a generic exponential function, except for the fact that an extra factor of n is included as well.

II. This graph also shows a steep rise near the input value of 96. The output moves in the order of 32^{nd} power of 10 while the input is just 100.

III. Upon the execution of this function, we get to know that vague values are obtained for input values that are greater than 1014. This is shown below.

```
1008    2765006530672019100000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1009    5535499185412831900000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1010    1108197061896325100000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1011    2218588573420167700000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1012    4441566046095370400000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1013    8891909890700810600000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1014    1780137537842176100000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000.00
1015    1.#J
1016    1.#J
1017    1.#J
1018    1.#J
1019    1.#J
1020    1.#J
1021    1.#J
1022    1.#J
1023    1.#J
1024    1.#J
1025    1.#J
```

[J] Function-10: $\log_2 n$



<u>Observations</u>-

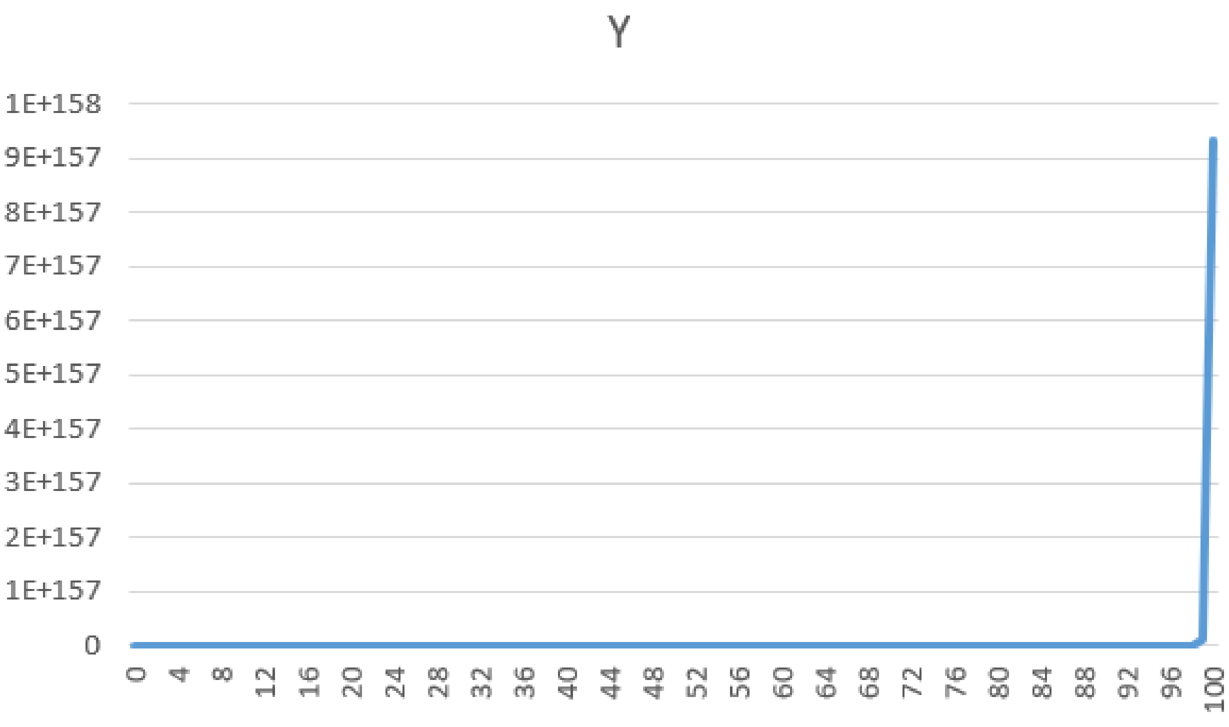
I. The graph of this function roughly imitates the one belonging to a generic logarithmic function having a base value greater than 1.

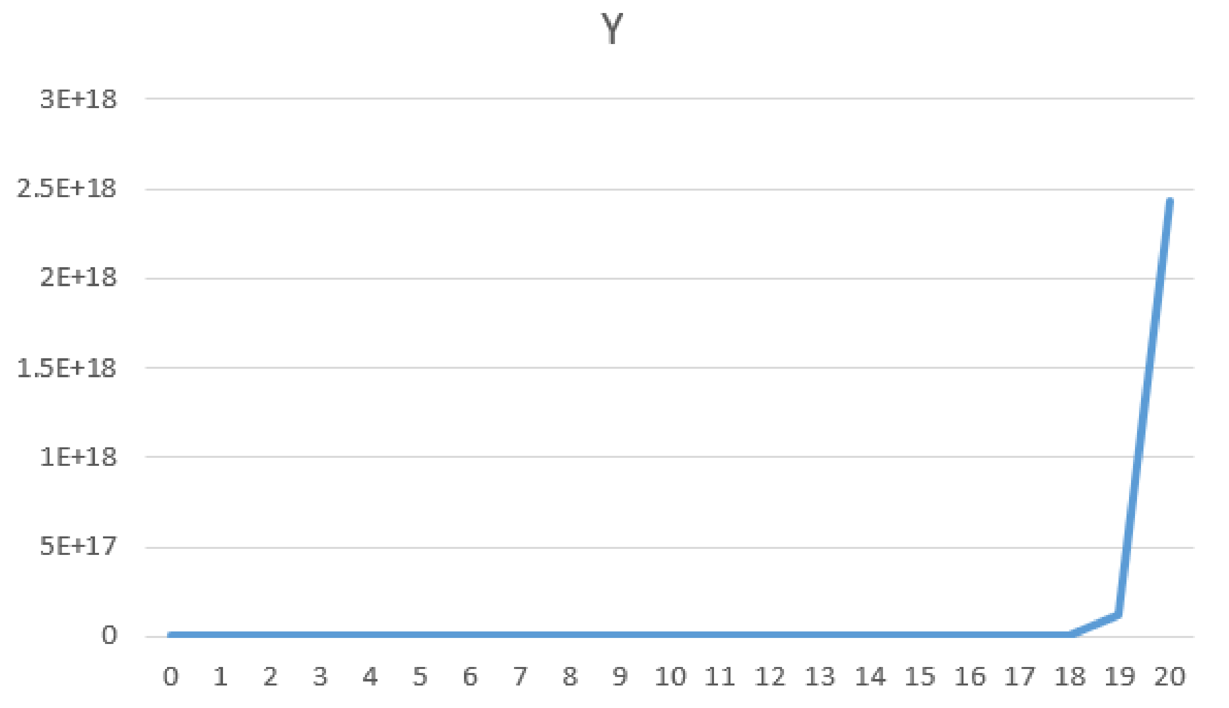
II. This function is undefined at the input value of 0, the output at which is shown as 0 in the graph as per the default behaviour of the graph in an excel file.

III. This function, being a logarithmic one, succeeds in providing proper outputs even when the inputs are in millions or billions.

```
1000045037      29.90
1000045038      29.90
1000045039      29.90
1000045040      29.90
1000045041      29.90
1000045042      29.90
1000045043      29.90
1000045044      29.90
1000045045      29.90
1000045046      29.90
1000045047      29.90
1000045048      29.90
1000045049      29.90
1000045050      29.90
1000045051      29.90
1000045052      29.90
1000045053      29.90
1000045054      29.90
1000045055      29.90
```

[K] Function-11: n!





<u>Observations</u>-

I. From both the above graphs that are plotted for the ranges 0 to 100 and 0 to 20 respectively, we find that even the factorial function has a rapid increase in its output when the input increases by a small value.

II. As observed in function-4, upon trying to calculate the factorials of inputs greater than 100, we find that after the value of 170, vague values are found in the output.

Given below is a glimpse of the excel file where all the x-y values for all the various functions were pasted.

Function-1		Function-2		Function-3		Function-4	
X	Y	X	Y	X	Y	X	Y
0	1	0	0	0	1.#QNB	0	0
1	1.5	1	1	1	-1.#INF	1	0
2	2.25	2	8	2	-0.3665	2	1
3	3.38	3	27	3	0.094	3	2.58
4	5.06	4	64	4	0.3266	4	4.58
5	7.59	5	125	5	0.4759	5	6.91
6	11.39	6	216	6	0.5832	6	9.49
7	17.09	7	343	7	0.6657	7	12.3
8	25.63	8	512	8	0.7321	8	15.3
9	38.44	9	729	9	0.7872	9	18.47
10	57.67	10	1000	10	0.834	10	21.79
11	86.5	11	1331	11	0.8746	11	25.25
12	129.75	12	1728	12	0.9102	12	28.84
13	194.62	13	2197	13	0.9419	13	32.54
14	291.93	14	2744	14	0.9704	14	36.34
15	437.89	15	3375	15	0.9962	15	40.25
16	656.84	16	4096	16	1.0198	16	44.25
17	985.26	17	4913	17	1.0414	17	48.34
18	1477.89	18	5832	18	1.0614	18	52.51
19	2216.84	19	6859	19	1.0799	19	56.76
20	3325.26	20	8000	20	1.0972	20	61.08
21	4987.89	21	9261	21	1.1133	21	65.47
22	7481.83	22	10648	22	1.1285	22	69.93
23	11222.74	23	12167	23	1.1428	23	74.45
24	16834.11	24	13824	24	1.1563	24	79.04
25	25251.17	25	15625	25	1.169	25	83.68
26	37876.75	26	17576	26	1.1811	26	88.38
27	56815.13	27	19683	27	1.1927	27	93.14
28	85222.69	28	21952	28	1.2036	28	97.94
29	127834	29	24389	29	1.2141	29	102.8
30	191751.1	30	27000	30	1.2241	30	107.71
31	287626.6	31	29791	31	1.2337	31	112.66
32	431439.9	32	32768	32	1.2429	32	117.66
33	647159.8	33	35937	33	1.2518	33	122.71
34	970739.7	34	39304	34	1.2603	34	127.8
35	1456110	35	42875	35	1.2685	35	132.92
36	2184164	36	46656	36	1.2763	36	138.09
37	3276247	37	50653	37	1.284	37	143.3
38	4914370	38	54872	38	1.2913	38	148.55
39	7371555	39	59319	39	1.2984	39	153.84
40	11057332	40	64000	40	1.3053	40	159.16
41	16585998	41	68921	41	1.312	41	164.52
42	24878998	42	74088	42	1.3185	42	169.91
43	37318497	43	79507	43	1.3247	43	175.34
44	55977745	44	85184	44	1.3308	44	180.79
45	83966617	45	91125	45	1.3368	45	186.29
46	1.26E+08	46	97336	46	1.3425	46	191.81
47	1.89E+08	47	103823	47	1.3481	47	197.36
48	2.83E+08	48	110592	48	1.3536	48	202.95
49	4.25E+08	49	117649	49	1.3589	49	208.56
50	6.38E+08	50	125000	50	1.3641	50	214.21
51	9.56E+08	51	132651	51	1.3691	51	219.88
52	1.43E+09	52	140608	52	1.374	52	225.58
53	2.15E+09	53	148877	53	1.3788	53	231.31

54	3.23E+09
55	4.84E+09
56	7.26E+09
57	1.09E+10
58	1.63E+10
59	2.45E+10
60	3.68E+10
61	5.52E+10
62	8.27E+10
63	1.24E+11
64	1.86E+11
65	2.79E+11
66	4.19E+11
67	6.28E+11
68	9.42E+11
69	1.41E+12
70	2.12E+12
71	3.18E+12
72	4.77E+12
73	7.16E+12
74	1.07E+13
75	1.61E+13
76	2.42E+13
77	3.62E+13
78	5.43E+13
79	8.15E+13
80	1.22E+14
81	1.83E+14
82	2.75E+14
83	4.13E+14
84	6.19E+14
85	9.28E+14
86	1.39E+15
87	2.09E+15
88	3.13E+15
89	4.7E+15
90	7.05E+15
91	1.06E+16
92	1.59E+16
93	2.38E+16
94	3.57E+16
95	5.35E+16
96	8.03E+16
97	1.2E+17
98	1.81E+17
99	2.71E+17
100	4.07E+17

54	157464
55	166375
56	175616
57	185193
58	195112
59	205379
60	216000
61	226981
62	238328
63	250047
64	262144
65	274625
66	287496
67	300763
68	314432
69	328509
70	343000
71	357911
72	373248
73	389017
74	405224
75	421875
76	438976
77	456533
78	474552
79	493039
80	512000
81	531441
82	551368
83	571787
84	592704
85	614125
86	636056
87	658503
88	681472
89	704969
90	729000
91	753571
92	778688
93	804357
94	830584
95	857375
96	884736
97	912673
98	941192
99	970299
100	1000000

54	1.3835
55	1.3881
56	1.3926
57	1.397
58	1.4013
59	1.4055
60	1.4096
61	1.4136
62	1.4176
63	1.4215
64	1.4252
65	1.429
66	1.4326
67	1.4362
68	1.4397
69	1.4432
70	1.4466
71	1.4499
72	1.4532
73	1.4564
74	1.4596
75	1.4627
76	1.4657
77	1.4688
78	1.4717
79	1.4746
80	1.4775
81	1.4803
82	1.4831
83	1.4859
84	1.4886
85	1.4913
86	1.4939
87	1.4965
88	1.499
89	1.5015
90	1.504
91	1.5065
92	1.5089
93	1.5113
94	1.5137
95	1.516
96	1.5183
97	1.5205
98	1.5228
99	1.525
100	1.5272

54	237.06
55	242.85
56	248.65
57	254.49
58	260.34
59	266.23
60	272.13
61	278.06
62	284.02
63	290
64	296
65	302.02
66	308.06
67	314.13
68	320.22
69	326.32
70	332.45
71	338.6
72	344.77
73	350.96
74	357.17
75	363.4
76	369.65
77	375.92
78	382.2
79	388.5
80	394.83
81	401.17
82	407.52
83	413.9
84	420.29
85	426.7
86	433.13
87	439.57
88	446.03
89	452.51
90	459
91	465.51
92	472.03
93	478.57
94	485.12
95	491.69
96	498.28
97	504.88
98	511.49
99	518.12
100	524.76

Function-5		Function-6		Function-7		Function-8	
X	Y	X	Y	X	Y	X	Y
0	1	0	1.#QNB	0	0	0	2
1	2.72	1	-1.#INF	1	1	1	4
2	7.39	2	0	2	2	2	16
3	20.09	3	0.6644	3	3	3	256
4	54.6	4	1	4	4	4	65536
5	148.41	5	1.2153	5	5	5	4.29E+09
6	403.43	6	1.3701	6	6	6	1.84E+19
7	1096.63	7	1.4892	7	7	7	3.4E+38
8	2980.96	8	1.585	8	8	8	1.16E+77
9	8103.08	9	1.6644	9	9	9	1.3E+154
10	22026.47	10	1.732	10	10	10	1.#J
11	59874.14	11	1.7905	11	11	11	1.#J
12	162754.8	12	1.842	12	12	12	1.#J
13	442413.4	13	1.8877	13	13	13	1.#J
14	1202604	14	1.9288	14	14	14	1.#J
15	3269017	15	1.966	15	15	15	1.#J
16	8886111	16	2	16	16	16	1.#J
17	24154953	17	2.0312	17	17	17	1.#J
18	65659969	18	2.06	18	18	18	1.#J
19	1.78E+08	19	2.0868	19	19	19	1.#J
20	4.85E+08	20	2.1117	20	20	20	1.#J
21	1.32E+09	21	2.135	21	21	21	1.#J
22	3.58E+09	22	2.1569	22	22	22	1.#J
23	9.74E+09	23	2.1775	23	23	23	1.#J
24	2.65E+10	24	2.1969	24	24	24	1.#J
25	7.2E+10	25	2.2153	25	25	25	1.#J
26	1.96E+11	26	2.2328	26	26	26	1.#J
27	5.32E+11	27	2.2494	27	27	27	1.#J
28	1.45E+12	28	2.2652	28	28	28	1.#J
29	3.93E+12	29	2.2804	29	29	29	1.#J
30	1.07E+13	30	2.2948	30	30	30	1.#J
31	2.9E+13	31	2.3087	31	31	31	1.#J
32	7.9E+13	32	2.3219	32	32	32	1.#J
33	2.15E+14	33	2.3347	33	33	33	1.#J
34	5.83E+14	34	2.3469	34	34	34	1.#J
35	1.59E+15	35	2.3588	35	35	35	1.#J
36	4.31E+15	36	2.3701	36	36	36	1.#J
37	1.17E+16	37	2.3811	37	37	37	1.#J
38	3.19E+16	38	2.3917	38	38	38	1.#J
39	8.66E+16	39	2.402	39	39	39	1.#J
40	2.35E+17	40	2.4119	40	40	40	1.#J
41	6.4E+17	41	2.4216	41	41	41	1.#J
42	1.74E+18	42	2.4309	42	42	42	1.#J
43	4.73E+18	43	2.44	43	43	43	1.#J
44	1.29E+19	44	2.4488	44	44	44	1.#J
45	3.49E+19	45	2.4573	45	45	45	1.#J
46	9.5E+19	46	2.4656	46	46	46	1.#J
47	2.58E+20	47	2.4737	47	47	47	1.#J
48	7.02E+20	48	2.4815	48	48	48	1.#J
49	1.91E+21	49	2.4892	49	49	49	1.#J
50	5.18E+21	50	2.4967	50	50	50	1.#J
51	1.41E+22	51	2.504	51	51	51	1.#J
52	3.83E+22	52	2.5111	52	52	52	1.#J
53	1.04E+23	53	2.518	53	53	53	1.#J

54	2.83E+23		54	2.5248		54	54		54	1.#J
55	7.69E+23		55	2.5314		55	55		55	1.#J
56	2.09E+24		56	2.5379		56	56		56	1.#J
57	5.69E+24		57	2.5442		57	57		57	1.#J
58	1.55E+25		58	2.5504		58	58		58	1.#J
59	4.2E+25		59	2.5565		59	59		59	1.#J
60	1.14E+26		60	2.5624		60	60		60	1.#J
61	3.1E+26		61	2.5682		61	61		61	1.#J
62	8.44E+26		62	2.5739		62	62		62	1.#J
63	2.29E+27		63	2.5795		63	63		63	1.#J
64	6.24E+27		64	2.585		64	64		64	1.#J
65	1.69E+28		65	2.5903		65	65		65	1.#J
66	4.61E+28		66	2.5956		66	66		66	1.#J
67	1.25E+29		67	2.6008		67	67		67	1.#J
68	3.4E+29		68	2.6058		68	68		68	1.#J
69	9.25E+29		69	2.6108		69	69		69	1.#J
70	2.52E+30		70	2.6157		70	70		70	1.#J
71	6.84E+30		71	2.6205		71	71		71	1.#J
72	1.86E+31		72	2.6253		72	72		72	1.#J
73	5.05E+31		73	2.6299		73	73		73	1.#J
74	1.37E+32		74	2.6345		74	74		74	1.#J
75	3.73E+32		75	2.639		75	75		75	1.#J
76	1.01E+33		76	2.6434		76	76		76	1.#J
77	2.76E+33		77	2.6477		77	77		77	1.#J
78	7.5E+33		78	2.652		78	78		78	1.#J
79	2.04E+34		79	2.6562		79	79		79	1.#J
80	5.54E+34		80	2.6604		80	80		80	1.#J
81	1.51E+35		81	2.6644		81	81		81	1.#J
82	4.09E+35		82	2.6685		82	82		82	1.#J
83	1.11E+36		83	2.6724		83	83		83	1.#J
84	3.03E+36		84	2.6763		84	84		84	1.#J
85	8.22E+36		85	2.6802		85	85		85	1.#J
86	2.24E+37		86	2.684		86	86		86	1.#J
87	6.08E+37		87	2.6877		87	87		87	1.#J
88	1.65E+38		88	2.6914		88	88		88	1.#J
89	4.49E+38		89	2.695		89	89		89	1.#J
90	1.22E+39		90	2.6986		90	90		90	1.#J
91	3.32E+39		91	2.7022		91	91		91	1.#J
92	9.02E+39		92	2.7057		92	92		92	1.#J
93	2.45E+40		93	2.7091		93	93		93	1.#J
94	6.66E+40		94	2.7125		94	94		94	1.#J
95	1.81E+41		95	2.7159		95	95		95	1.#J
96	4.92E+41		96	2.7192		96	96		96	1.#J
97	1.34E+42		97	2.7224		97	97		97	1.#J
98	3.64E+42		98	2.7257		98	98		98	1.#J
99	9.89E+42		99	2.7289		99	99		99	1.#J
100	2.69E+43		100	2.732		100	100		100	1.#J

Function-9	
X	Y
0	0
1	2
2	8
3	24
4	64
5	160
6	384
7	896
8	2048
9	4608
10	10240
11	22528
12	49152
13	106496
14	229376
15	491520
16	1048576
17	2228224
18	4718592
19	9961472
20	20971520
21	44040192
22	92274688
23	1.93E+08
24	4.03E+08
25	8.39E+08
26	1.74E+09
27	3.62E+09
28	7.52E+09
29	1.56E+10
30	3.22E+10
31	6.66E+10
32	1.37E+11
33	2.83E+11
34	5.84E+11
35	1.2E+12
36	2.47E+12
37	5.09E+12
38	1.04E+13
39	2.14E+13
40	4.4E+13
41	9.02E+13
42	1.85E+14
43	3.78E+14
44	7.74E+14
45	1.58E+15
46	3.24E+15
47	6.61E+15
48	1.35E+16
49	2.76E+16
50	5.63E+16
51	1.15E+17
52	2.34E+17
53	4.77E+17

Function-10	
X	Y
0	-1.#J
1	0
2	1
3	1.58
4	2
5	2.32
6	2.58
7	2.81
8	3
9	3.17
10	3.32
11	3.46
12	3.58
13	3.7
14	3.81
15	3.91
16	4
17	4.09
18	4.17
19	4.25
20	4.32
21	4.39
22	4.46
23	4.52
24	4.58
25	4.64
26	4.7
27	4.75
28	4.81
29	4.86
30	4.91
31	4.95
32	5
33	5.04
34	5.09
35	5.13
36	5.17
37	5.21
38	5.25
39	5.29
40	5.32
41	5.36
42	5.39
43	5.43
44	5.46
45	5.49
46	5.52
47	5.55
48	5.58
49	5.61
50	5.64
51	5.67
52	5.7
53	5.73

Function-11	
X	Y
0	1
1	1
2	2
3	6
4	24
5	120
6	720
7	5040
8	40320
9	362880
10	3628800
11	39916800
12	4.79E+08
13	6.23E+09
14	8.72E+10
15	1.31E+12
16	2.09E+13
17	3.56E+14
18	6.4E+15
19	1.22E+17
20	2.43E+18
21	5.11E+19
22	1.12E+21
23	2.59E+22
24	6.2E+23
25	1.55E+25
26	4.03E+26
27	1.09E+28
28	3.05E+29
29	8.84E+30
30	2.65E+32
31	8.22E+33
32	2.63E+35
33	8.68E+36
34	2.95E+38
35	1.03E+40
36	3.72E+41
37	1.38E+43
38	5.23E+44
39	2.04E+46
40	8.16E+47
41	3.35E+49
42	1.41E+51
43	6.04E+52
44	2.66E+54
45	1.2E+56
46	5.5E+57
47	2.59E+59
48	1.24E+61
49	6.08E+62
50	3.04E+64
51	1.55E+66
52	8.07E+67
53	4.27E+69

54	9.73E+17
55	1.98E+18
56	4.04E+18
57	8.21E+18
58	1.67E+19
59	3.4E+19
60	6.92E+19
61	1.41E+20
62	2.86E+20
63	5.81E+20
64	1.18E+21
65	2.4E+21
66	4.87E+21
67	9.89E+21
68	2.01E+22
69	4.07E+22
70	8.26E+22
71	1.68E+23
72	3.4E+23
73	6.89E+23
74	1.4E+24
75	2.83E+24
76	5.74E+24
77	1.16E+25
78	2.36E+25
79	4.78E+25
80	9.67E+25
81	1.96E+26
82	3.97E+26
83	8.03E+26
84	1.62E+27
85	3.29E+27
86	6.65E+27
87	1.35E+28
88	2.72E+28
89	5.51E+28
90	1.11E+29
91	2.25E+29
92	4.56E+29
93	9.21E+29
94	1.86E+30
95	3.76E+30
96	7.61E+30
97	1.54E+31
98	3.11E+31
99	6.27E+31
100	1.27E+32

54	5.75
55	5.78
56	5.81
57	5.83
58	5.86
59	5.88
60	5.91
61	5.93
62	5.95
63	5.98
64	6
65	6.02
66	6.04
67	6.07
68	6.09
69	6.11
70	6.13
71	6.15
72	6.17
73	6.19
74	6.21
75	6.23
76	6.25
77	6.27
78	6.29
79	6.3
80	6.32
81	6.34
82	6.36
83	6.38
84	6.39
85	6.41
86	6.43
87	6.44
88	6.46
89	6.48
90	6.49
91	6.51
92	6.52
93	6.54
94	6.55
95	6.57
96	6.58
97	6.6
98	6.61
99	6.63
100	6.64

54	2.31E+71
55	1.27E+73
56	7.11E+74
57	4.05E+76
58	2.35E+78
59	1.39E+80
60	8.32E+81
61	5.08E+83
62	3.15E+85
63	1.98E+87
64	1.27E+89
65	8.25E+90
66	5.44E+92
67	3.65E+94
68	2.48E+96
69	1.71E+98
70	1.2E+100
71	8.5E+101
72	6.1E+103
73	4.5E+105
74	3.3E+107
75	2.5E+109
76	1.9E+111
77	1.5E+113
78	1.1E+115
79	8.9E+116
80	7.2E+118
81	5.8E+120
82	4.8E+122
83	3.9E+124
84	3.3E+126
85	2.8E+128
86	2.4E+130
87	2.1E+132
88	1.9E+134
89	1.7E+136
90	1.5E+138
91	1.4E+140
92	1.2E+142
93	1.2E+144
94	1.1E+146
95	1E+148
96	9.9E+149
97	9.6E+151
98	9.4E+153
99	9.3E+155
100	9.3E+157

Conclusion:

By performing this experiment, I was able to observe the difference in the various functions that were implemented. I was also able to understand the procedure of plotting a graph from the obtained data using Microsoft excel.