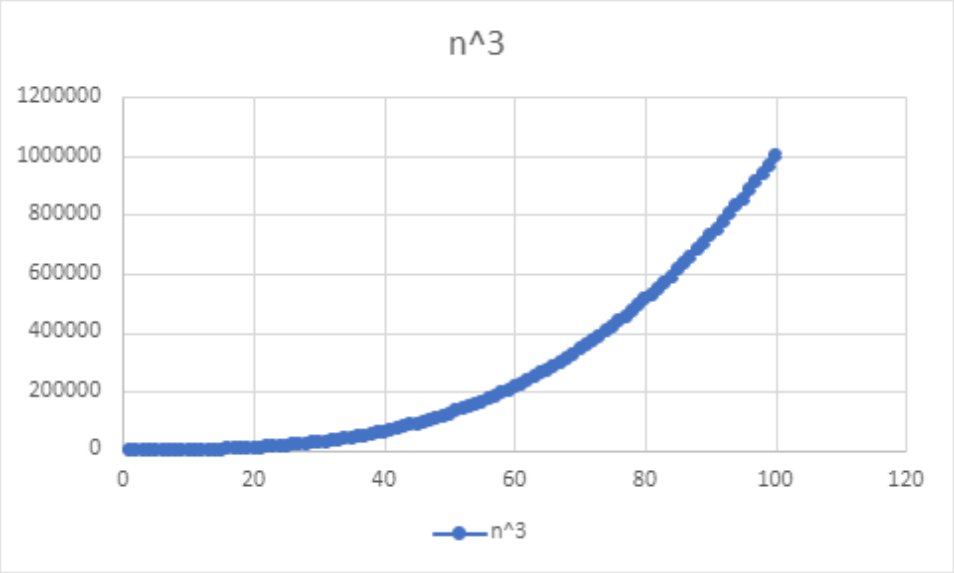
NAME:	Harshith Kunder
UID:	2021300068
SUBJECT	Design and Analysis of Algorithms.
EXPERIMENT NO :	1 A.
PROBLEM STATEMENT 1:	To implement the various functions e.g. linear, non-linear,quadratic, exponential etc.

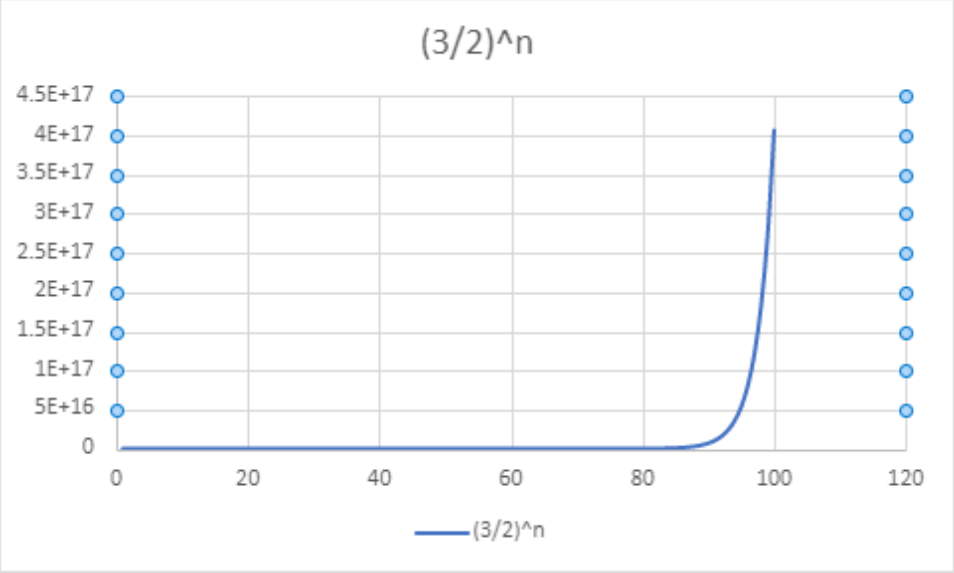
Program:

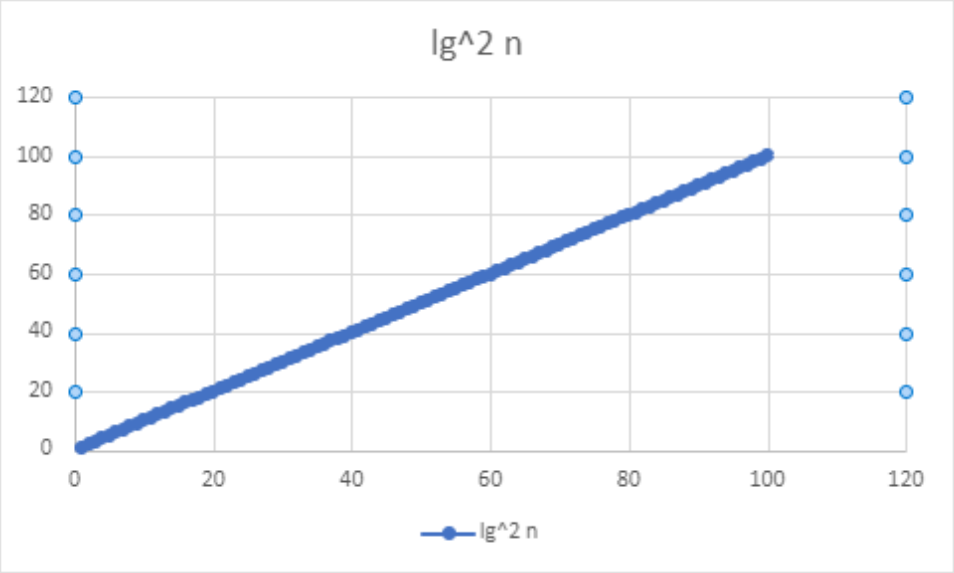
```
#include<bits/stdc++.h>
using namespace std;
using ll=long long;

float firstFn(float n){
  float ans=powf((float)3/2,n);
  return ans;
}
float secondFn(float n){
  return powf(n,3);
}
float thirdFn(float n){
  return powf(2,(float)log2(n));
}
double fourthFn(int n){
  int factorial=1;
  for (int a=1;a<=n;a++) {
    factorial=factorial*a;
  }
  return log2(factorial);
}
double fifthFn(float n){
  double x=powf(2,n);
  return powf(2,x);
}
float sixthFn(float n){
```

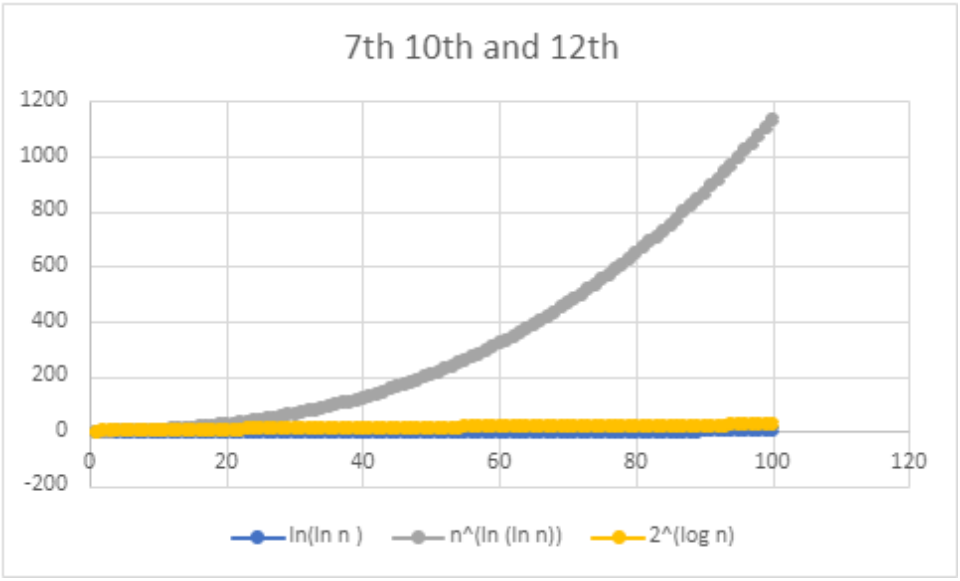
```
  return powf(n,(float)1/log2(n));
}
float seventhFn(float n){
  return log(log(n));
}
float ninthFn(float n){
   return (float)n*powf(2,n);
}
float tenthFn(float n){
  return powf(n,log(log(n)));
}
float twelfthFn(float n){
  return powf(2,log(n));
}


int main()
{
  for(int n=1;n<=100;n++){
    //cout<<(float)twelfth(n)<<" ";
  }

    return 0;
}
```

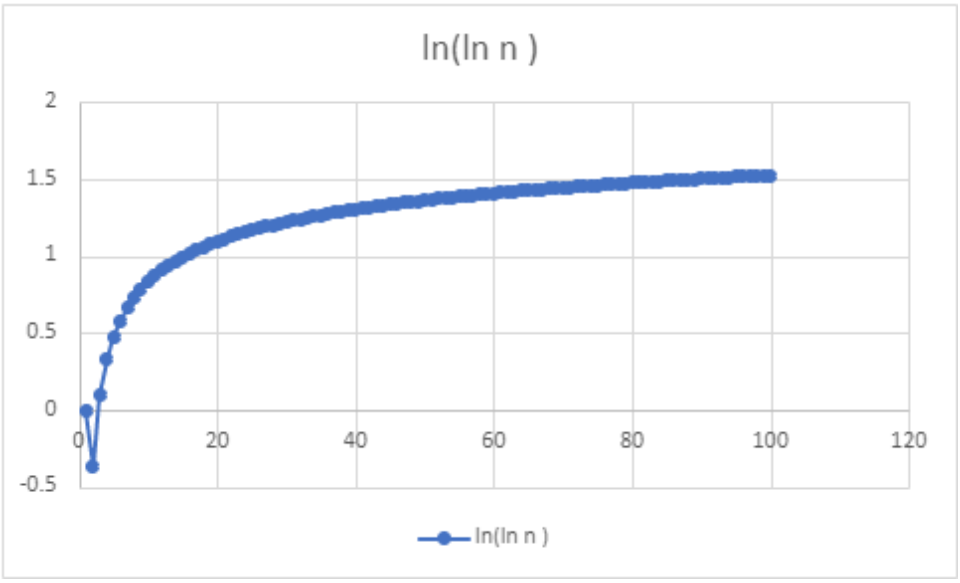
Graphs:

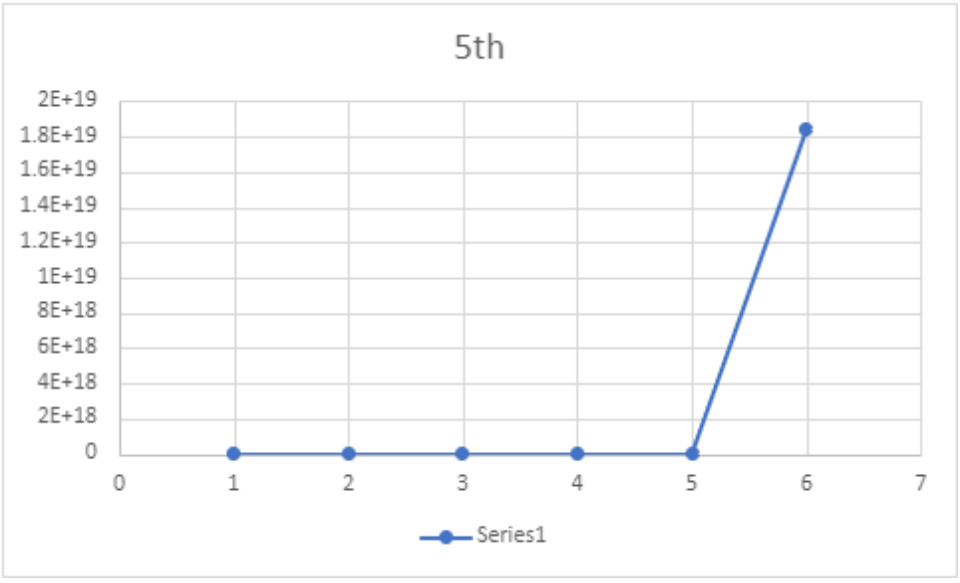

n^3
1200000
1000000
800000
600000
400000
200000
0
0
20
40
60
80
100
120
n^3

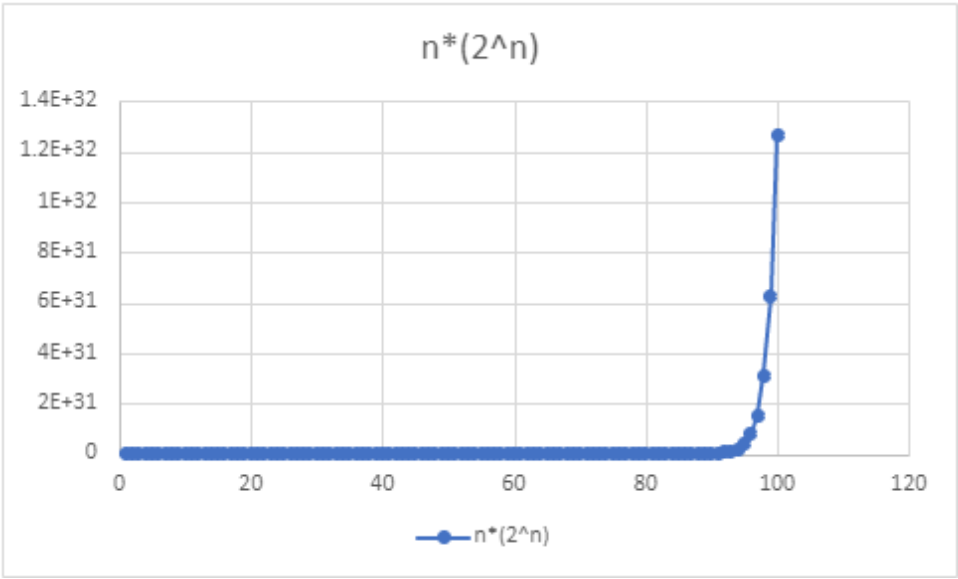

(3/2)^n
4.5E+17
4E+17
3.5E+17
3E+17
2.5E+17
2E+17
1.5E+17
1E+17
5E+16
0
0
20
40
60
80
100
120
(3/2)^n


lg^2 n
120
100
80
60
40
20
0
0
20
40
60
80
100
120
lg^2 n


7th 10th and 12th
1200
1000
800
600
400
200
0
-200
0
20
40
60
80
100
120
ln(ln n)
n^(ln (ln n))
2^(log n)


ln(ln n)
2
1.5
1
0.5
0
-0.5
0
20
40
60
80
100
120
ln(ln n)


5th
2E+19
1.8E+19
1.6E+19
1.4E+19
1.2E+19
1E+19
8E+18
6E+18
4E+18
2E+18
0
0
1
2
3
4
5
6
7
Series1



Observation Table:

n	(3/2)^n	n^3	lg^2 n	lg(n!)	2^(2^n)	ln(ln n)	n*(2^n)	n^(ln (ln n))	2^(log n)
1	1.5	1	1	0	4	#NAME?	2	1	1
2	2.25	8	2	1	16	-0.366513	8	0.775655	1.61681
3	3.375	27	3	2.58496	256	0.094047	24	1.10885	2.14149
4	5.0625	64	4	4.58496	65536	0.326634	64	1.57273	2.61406
5	7.59375	125	5	6.90689	4.29E+09	0.475885	160	2.15095	3.05133
6	11.3906	216	6	9.49185	1.84E+19	0.583198	384	2.84326	3.46237
7	17.0859	343	7	12.2992	inf	0.66573	896	3.65264	3.85281
8	25.6289	512	8	15.2992	inf	0.732099	2048	4.58302	4.22644
9	38.4434	729	9	18.4691	inf	0.787195	4608	5.63865	4.58596
10	57.665	1000	10	21.7911	inf	0.834032	10240	6.8239	4.93341
11	86.4976	1331	11	25.2505	inf	0.874591	22528	8.14316	5.27034
12	129.746	1728	12	28.8355	inf	0.910235	49152	9.60083	5.59798
13	194.62	2197	13	30.8475	inf	0.941939	106496	11.2012	5.91734

14	291.929	2744	14	30.2523	inf	0.970422	229376	12.9487	6.22925
15	437.894	3375	15	30.9005	inf	0.996229	491520	14.8476	6.53438
16	656.841	4096	16	30.9004	inf	1.01978	1.05E+06	16.902	6.83333
17	985.261	4913	17	nan	inf	1.04141	2.23E+06	19.1163	7.1266
18	1477.89	5832	18	nan	inf	1.06139	4.72E+06	21.4945	7.41462
19	2216.84	6859	19	26.7082	inf	1.07992	9.96E+06	24.0408	7.69776
20	3325.26	8000	20	nan	inf	1.09719	2.10E+07	26.7593	7.97637
21	4987.89	9261	21	nan	inf	1.11334	4.40E+07	29.6541	8.25073
22	7481.83	10648	22	nan	inf	1.12851	9.23E+07	32.7292	8.52112
23	11222.7	12167	23	29.6839	inf	1.14279	1.93E+08	35.9886	8.78775
24	16834.1	13824	24	nan	inf	1.15627	4.03E+08	39.4362	9.05085
25	25251.2	15625	25	30.9513	inf	1.16903	8.39E+08	43.0762	9.31061
26	37876.8	17576	26	nan	inf	1.18114	1.74E+09	46.9122	9.5672
27	56815.1	19683	27	30.4676	inf	1.19266	3.62E+09	50.9484	9.82078
28	85222.7	21952	28	nan	inf	1.20363	7.52E+09	55.1885	10.0715
29	127834	24389	29	nan	inf	1.21411	1.56E+10	59.6365	10.3195
30	191751	27000	30	30.3923	inf	1.22413	3.22E+10	64.2961	10.5648
31	287627	29791	31	29.4594	inf	1.23372	6.66E+10	69.1713	10.8077
32	431440	32768	32	nan	inf	1.24293	1.37E+11	74.2658	11.0482
33	647160	35937	33	nan	inf	1.25176	2.83E+11	79.5833	11.2864
34	970740	39304	34	#NAME?	inf	1.26027	5.84E+11	85.1278	11.5223

							1		
35	1.46E+06	42875	35	#NAME?	inf	1.26845	1.20E+12	90.9029	11.7562
36	2.18E+06	46656	36	#NAME?	inf	1.27635	2.47E+12	96.9124	11.988
37	3.28E+06	50653	37	#NAME?	inf	1.28396	5.09E+12	103.16	12.2178
38	4.91E+06	54872	38	#NAME?	inf	1.29132	1.04E+13	109.65	12.4458
39	7.37E+06	59319	39	#NAME?	inf	1.29844	2.14E+13	116.385	12.6719
40	1.11E+07	64000	40	#NAME?	inf	1.30532	4.40E+13	123.369	12.8962
41	1.66E+07	68921	41	#NAME?	inf	1.31199	9.02E+13	130.606	13.1189
42	2.49E+07	74088	42	#NAME?	inf	1.31846	1.85E+14	138.1	13.3398
43	3.73E+07	79507	43	#NAME?	inf	1.32474	3.78E+14	145.854	13.5592
44	5.60E+07	85184	44	#NAME?	inf	1.33083	7.74E+14	153.872	13.777
45	8.40E+07	91125	45	#NAME?	inf	1.33675	1.58E+15	162.157	13.9933
46	1.26E+08	97336	46	#NAME?	inf	1.34251	3.24E+15	170.714	14.2081
47	1.89E+08	103823	47	#NAME?	inf	1.34811	6.61E+15	179.545	14.4215
48	2.83E+08	110592	48	#NAME?	inf	1.35356	1.35E+16	188.655	14.6335
49	4.25E+08	117649	49	#NAME?	inf	1.35888	2.76E+16	198.047	14.8441
50	6.38E+08	125000	50	#NAME?	inf	1.36405	5.63E+16	207.725	15.0535
51	9.56E+08	132651	51	#NAME?	inf	1.3691	1.15E+17	217.691	15.2615
52	1.43E+09	140608	52	#NAME?	inf	1.37403	2.34E+17	227.951	15.4683
53	2.15E+09	148877	53	#NAME?	inf	1.37884	4.77E+17	238.506	15.6739

54	3.23E+09	157464	54	#NAME?	inf	1.38354	9.73E+17	249.362	15.8783
55	4.84E+09	166375	55	#NAME?	inf	1.38813	1.98E+18	260.521	16.0815
56	7.26E+09	175616	56	#NAME?	inf	1.39261	4.04E+18	271.987	16.2836
57	1.09E+10	185193	57	#NAME?	inf	1.397	8.21E+18	283.764	16.4846
58	1.63E+10	195112	58	#NAME?	inf	1.40129	1.67E+19	295.854	16.6846
59	2.45E+10	205379	59	#NAME?	inf	1.40549	3.40E+19	308.262	16.8834
60	3.68E+10	216000	60	#NAME?	inf	1.40961	6.92E+19	320.992	17.0813
61	5.52E+10	226981	61	#NAME?	inf	1.41364	1.41E+20	334.046	17.2781
62	8.27E+10	238328	62	#NAME?	inf	1.41758	2.86E+20	347.428	17.474
63	1.24E+11	250047	63	#NAME?	inf	1.42145	5.81E+20	361.141	17.6688
64	1.86E+11	262144	64	#NAME?	inf	1.42525	1.18E+21	375.191	17.8628
65	2.79E+11	274625	65	#NAME?	inf	1.42897	2.40E+21	389.578	18.0558
66	4.19E+11	287496	66	#NAME?	inf	1.43262	4.87E+21	404.308	18.2479
67	6.28E+11	300763	67	#NAME?	inf	1.4362	9.89E+21	419.383	18.4391
68	9.42E+11	314432	68	#NAME?	inf	1.43972	2.01E+22	434.807	18.6294
69	1.41E+12	328509	69	#NAME?	inf	1.44317	4.07E+22	450.584	18.8188
70	2.12E+12	343000	70	#NAME?	inf	1.44656	8.26E+22	466.717	19.0075
71	3.18E+12	357911	71	#NAME?	inf	1.4499	1.68E+23	483.21	19.1953
72	4.77E+12	373248	72	#NAME?	inf	1.45317	3.40E+23	500.065	19.3823
73	7.16E+11	389017	73	#NAME?	inf	1.45639	6.89E+22	517.287	19.5685

	2						3		
74	1.07E+13	405224	74	#NAME?	inf	1.45956	1.40E+24	534.879	19.7539
75	1.61E+13	421875	75	#NAME?	inf	1.46267	2.83E+24	552.844	19.9385
76	2.42E+13	438976	76	#NAME?	inf	1.46574	5.74E+24	571.186	20.1224
77	3.62E+13	456533	77	#NAME?	inf	1.46875	1.16E+25	589.908	20.3056
78	5.43E+13	474552	78	#NAME?	inf	1.47172	2.36E+25	609.015	20.488
79	8.15E+13	493039	79	#NAME?	inf	1.47464	4.78E+25	628.508	20.6697
80	1.22E+14	512000	80	#NAME?	inf	1.47751	9.67E+25	648.391	20.8507
81	1.83E+14	531441	81	#NAME?	inf	1.48034	1.96E+26	668.668	21.0311
82	2.75E+14	551368	82	#NAME?	inf	1.48313	3.97E+26	689.343	21.2107
83	4.13E+14	571787	83	#NAME?	inf	1.48588	8.03E+26	710.419	21.3896
84	6.19E+14	592704	84	#NAME?	inf	1.48858	1.62E+27	731.899	21.5679
85	9.28E+14	614125	85	#NAME?	inf	1.49125	3.29E+27	753.787	21.7456
86	1.39E+15	636056	86	#NAME?	inf	1.49388	6.65E+27	776.085	21.9226
87	2.09E+15	658503	87	#NAME?	inf	1.49647	1.35E+28	798.799	22.099
88	3.13E+15	681472	88	#NAME?	inf	1.49903	2.72E+28	821.93	22.2747
89	4.70E+15	704969	89	#NAME?	inf	1.50155	5.51E+28	845.482	22.4499
90	7.05E+15	729000	90	#NAME?	inf	1.50404	1.11E+29	869.459	22.6244
91	1.06E+16	753571	91	#NAME?	inf	1.50649	2.25E+29	893.865	22.7984
92	1.59E+16	778688	92	#NAME?	inf	1.50891	4.56E+29	918.702	22.9717

93	2.38E+16	804357	93	#NAME?	inf	1.5113	9.21E+29	943.973	23.1445
94	3.57E+16	830584	94	#NAME?	inf	1.51365	1.86E+30	969.683	23.3168
95	5.35E+16	857375	95	#NAME?	inf	1.51598	3.76E+30	995.835	23.4884
96	8.03E+16	884736	96	#NAME?	inf	1.51828	7.61E+30	1022.43	23.6595
97	1.20E+17	912673	97	#NAME?	inf	1.52054	1.54E+31	1049.48	23.8301
98	1.81E+17	941192	98	#NAME?	inf	1.52278	3.11E+31	1076.97	24.0001
99	2.71E+17	970299	99	#NAME?	inf	1.52499	6.27E+31	1104.93	24.1696
100	4.07E+17	1.00E+06	100	#NAME?	inf	1.52718	1.27E+32	1133.34	24.3385

Algorithm:	**Factorial:** Procedure fact(num) 1. until num=1 fact = fact*(num-1) 2. Print fact 3. end procedure
Conclusion:	By Performing the above experiment I understood how to create and analyze different functions.