Sardar Patel Institute of Technology

Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India

(Autonomous College Affiliated to University of Mumbai)

PRACTICAL NO. 1	DESIGN AND ANALYSIS OF ALGORITHMS
NAME	VEDANTI ANIL WADATKAR
UID	2021700072
BATCH	D4
PROBLEM STATEMENT	**Problem Definition & Assumptions** – For this experiment, you have to implement at least 10 functions from the following list. $(\frac{3}{2})^n$ n^3 $\lg^2 n$ $\lg(n!)$ 2^{2^n} $n^{1/\lg n}$ $\ln\ln n$ $\lg n$ $n \cdot 2^n$ $n^{\lg\lg n}$ $\ln n$ $2^{\lg n}$ $2^{\lg n}$ $(\lg n)^{\lg n}$ e^n $(\lg n)!$ $(\sqrt{2})^{\lg n}$ $\sqrt{\lg n}$ $\lg(\lg n)$ $2^{\sqrt{2\lg n}}$ n 2^n $n\lg n$ $2^{2^{n+1}}$
PROGRAM CODE	#include <stdio.h> #include<math.h> int n ; float f1(int n) { return pow(1.5,n) ;} float f2(int n) { return pow(n,3) ;} float f3(int n) { return pow(2,log(n));} float f4(int n) { return pow(log(n),log(n));} float f5(int n) { return (n)*log(n);} float f6(int n) { return log(log(n)) ;} float f7(int n) { return log(n);} float f8(int n) { return pow(2,n);} float f9(int n) { return pow(2,pow(n,n));} float f10(int n)

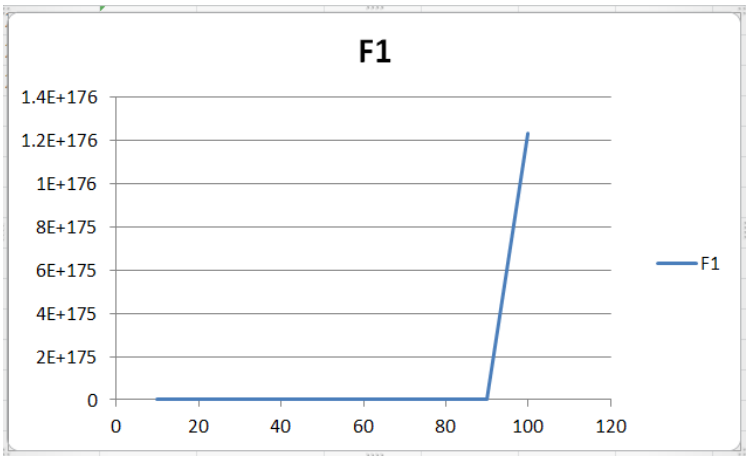
```
 { return exp(n) ;}


 int main()
{   int n[] = {10,20,30,40,50,60,70,80,90,100};
  int i;
  float ans1,ans2,ans3,ans4,ans5,ans6,ans7,ans8,ans9,ans10;
  printf("X   |   f1   | f2 | f3 | f4 | f5 | f6 | f7 | f8 | f9 |
f10  \n");
  printf("x=10 |");
  for(i=1; i<=10;i++)
{ ans1 = (f1(n[i]));
    printf ("%.2f|", ans1 );}
    printf("\n");
      printf("x=20 |");
    for(i=1; i<=10;i++)
{
    ans2 = (f2(n[i]));
printf ("%.2f|", ans2 );}

  printf("\n");
    printf("x=30 |");
 for(i=1; i<=10;i++)
{
    ans3 = (f3(n[i]));
printf ("%.2f|", ans3 );}
  printf("\n");
    printf("x=40 |");
 for(i=1; i<=10;i++)
{

ans4 = (f4(n[i]));
printf ("%.2f|", ans4);}
  printf("\n");
    printf("x=50 |");
 for(i=1; i<=10;i++)
{
    ans5 = (f5(n[i]));
printf ("%.2f|", ans5 );}
      printf("\n");
        printf("x=60 |");
     for(i=1; i<=10;i++)
{
ans6 = (f6(n[i]));
printf ("%.2f|", ans6 );}
```

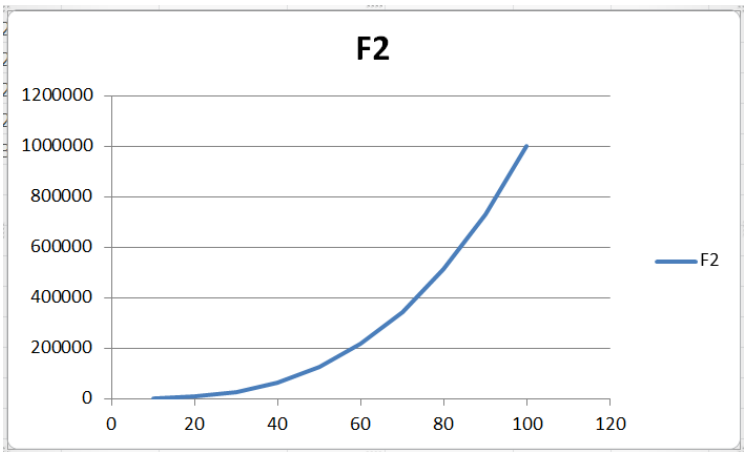
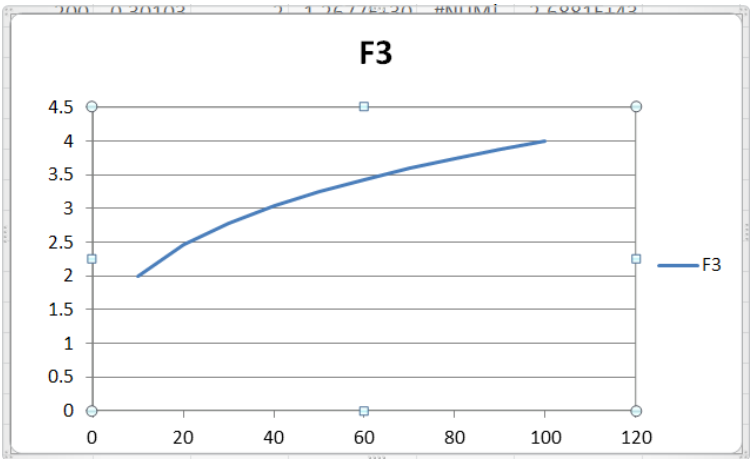
```
 printf("\n");
   printf("x=70 |");
 for(i=1; i<=10;i++)

{ans7 = (f7(n[i]));
printf ("%.2f|", ans7 );}
  printf("\n");
    printf("x=80 |");
 for(i=1; i<=10;i++)
{
ans8 = (f8(n[i]));
    printf ("%.2f|", ans8);}
       printf("\n");
         printf("x=90 |");
     for(i=1; i<=10;i++)
{
ans9 = (f9(n[i]));
printf ("%.2f  |", ans9 );}
   printf("\n");
     printf("x=100|");
 for(i=1; i<=10;i++)
{
ans10 = (f10(n[i]));
printf ("%.2f|", ans10);}

   return 0;
}
```

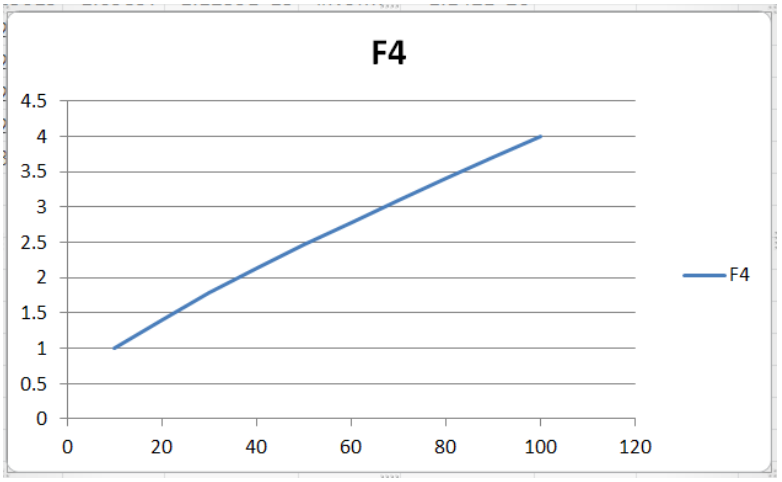
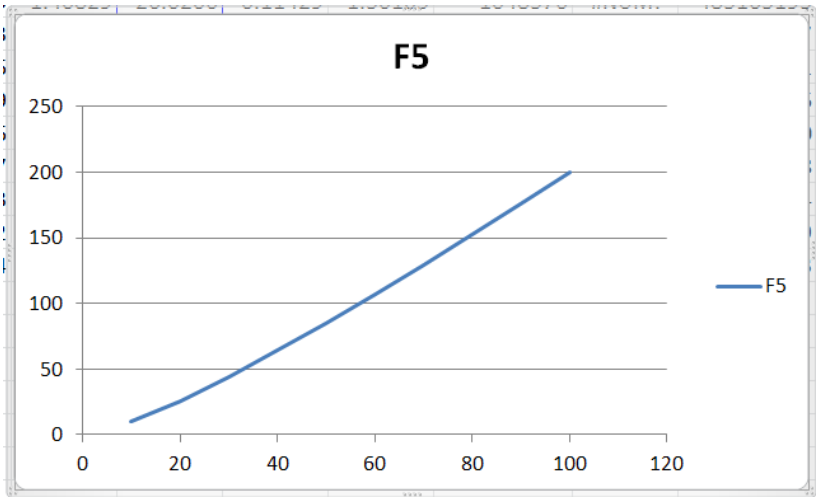
GRAPHICAL ANALYSIS

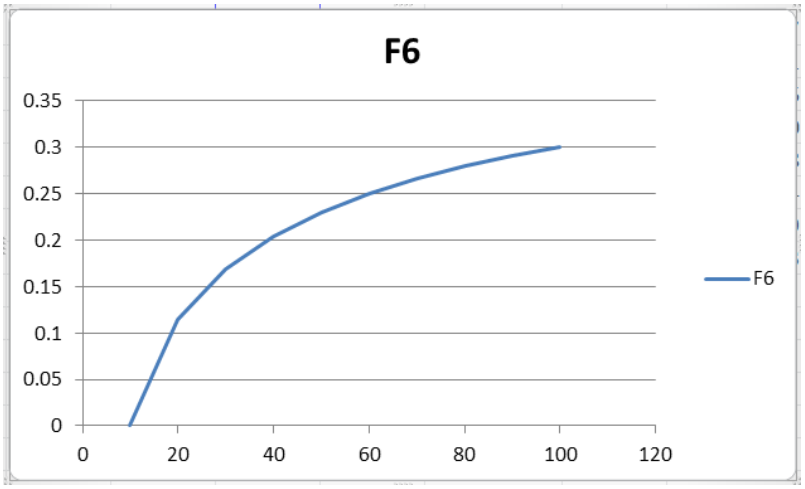
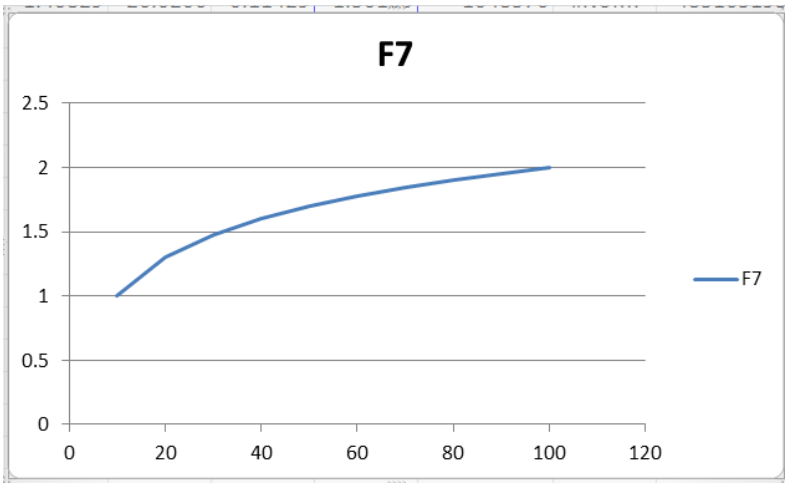
1)

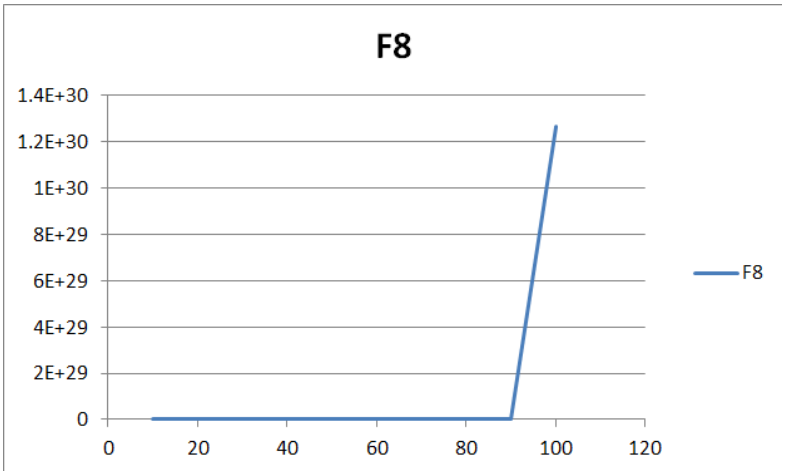
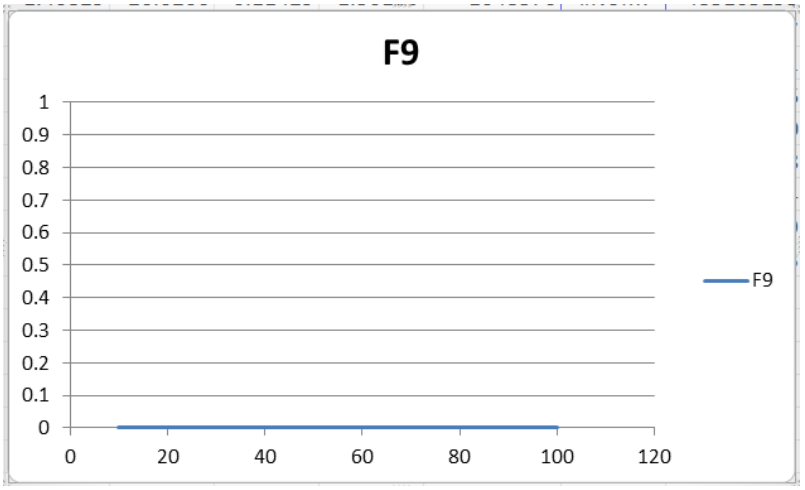
pow(1.5,n)

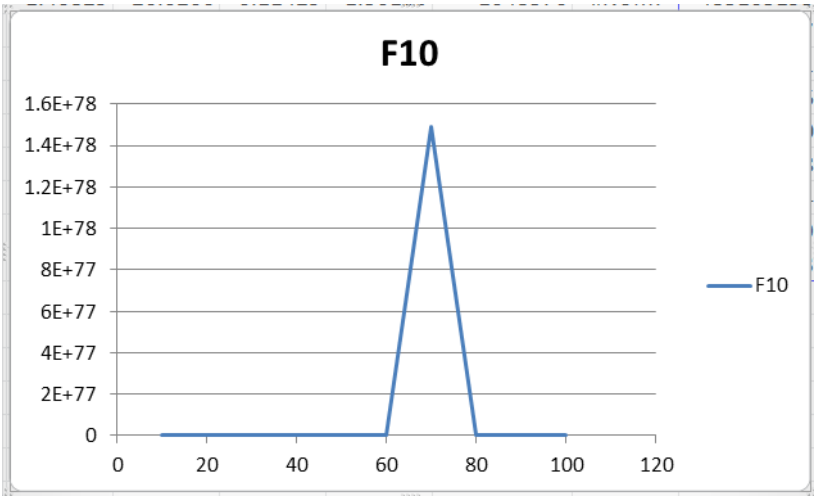


	ANALYSIS :- It increases steadily up to 80 and then shows sudden increase after that. It has recurring values in its answer so the graph barely shows any strong deviation.
2) **pow(n,3)**	
	ANALYSIS :- It is a cubic curve. It is symmetric curve. At every value of x it gives value of its cube. it is a increasing curve.
3) **pow(2,log(n))**	

	ANALYSIS :- The graph started and ended with definite value that is 2 and 4 for n value of 10 and 100 , respectively. And between them the graph is increasing. It concludes that for multiples of 10 the graph has definite values.
4) **pow(log(n),log(n))**	
	ANALYSIS :- This function has a steadily increasing graph. It starts with a finite value of 1 for n being 10. And for 100 , it also gives finite value that is 4. Between them the graph is increasing.
5) **(n)*log(n)**	

	ANALYSIS :- This function has a increasing graph bending downwards. As n gets multiplied at each step the value of the function also increases accordingly.
6) **log(log(n))**	
	ANALYSIS :- The graph is somewhat parabolic for this function. The function gives value less than 1 for every value under 100 starting from zero for n value of 10.
7) **log(n)**	

	ANALYSIS :- The values of this function strictly lies between 1 to 2. It is also a increasing graph.and it looks somewhat parabolic.
8) **pow(2,n)**	
	ANALYSIS :- This graph is a straight line graph due to high recurring values in the answer.
9) **pow(2,pow(n,n))**	

	ANALYSIS :- It is a straight line graph having value zero for all the values of n.
10) **Exp(n)**	
	ANALYSIS :- The function gives zero value up to 60 after which w observe a sudden rise in the graph up to 80 after that again the graph gives zero value.