

Numerical Methods Lab
Code: M(CS)491
Contact: 2P
Credit:1

1. Newton's forward Interpolation

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
int n,i,j;
float m,x[20],y[20],h,u,s,p=1,f[30][30];
clrscr();
printf("\nenter n(no. of (data-1))");
scanf("%d",&n);
printf("\n\n\enter the value of x1,x2,x3,.....,xn");
for(i=0;i<=n;i++)
{
scanf("%f",&x[i]);
}
printf("\n\n\enter the value of y1,y2,y3,.....,yn");
for(i=0;i<=n;i++)
{
scanf("%f",&y[i]);
}
h=x[1]-x[0];
printf("\n\n\enter the interpolating pt. x");
scanf("%f",&m);
u=(m-x[0])/h;
s=y[0];
for(j=0;j<=n;j++)
{
f[0][j]=y[j];
}
for(i=1;i<=n;i++)
{
for(j=0;j<=n-i;j++)
{
f[i][j]=f[i-1][j+1]-f[i-1][j];
}
}
for(i=1;i<=n;i++)
{
p=(p*(u-i+1))/i;
```

```

s=s+p*f[i][0];
}
printf("\n\n\tf(%6.4f)=%6.4f",m,s);
getch();
}

```

2. Newton's Backward Interpolation:

```

#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
int n,i,j;
float u, x, xn, h, sum=0, p=1, y[10][10];
clrscr();
printf("\n enter n, x, xn, h");
scanf("%d%f%f%f",&n,&x,&xn,&h);
printf("\n enter the value of y");
for(i=0;i<=n-1;i++)
scanf("%f", &y[0][i]);
for(i=1;i<=n-1;i++)
{
for(j=1;j<n;j++)
y[i][j]=y[i-1][j]-y[i-1][j-1];
}
u=(x-xn)/h;
for(i=1;i<=n-1;i++)
{
p=p*((u+i-1)/i);
sum=sum+(p*y[i][n-1]);
}
sum=sum+y[0][n-1];
printf("%f",sum);
getch();
}

```

3. Lagrange's Interpolation:

```

#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
clrscr();
int i,j,n;
float x[30],y[30],sum,xp,a,b;

```

```

printf("\n LAGRANGE METHOD");
printf("\n Enter the value of n:");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("\n Enter the values of x%d",i);
scanf("%f",&x[i]);
}
for(i=0;i<n;i++)
{
printf("\nEnter the values of y%d ",i);
scanf("%f",&y[i]);
}
printf("\nENTER THE VALUE OF INTERPOLATING POINT Xp ");
scanf("%f",&xp);
sum=0;
for(i=0;i<n;i++)
{
a=1;
b=1;
for(j=0;j<n;j++)
{
if(i!=j)
{
a=a*(xp-x[j]);
b=b*(x[i]-x[j]);
}
}
sum=sum+((a/b)*y[i]);
}
printf("THE VALUE OF f(x) is = %f",sum);
getch();
}

```

4. Newton Raphson Method

```

#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
float x0,xnew,eps;
clrscr();
printf("enter the value of x0 and eps\n");
scanf("%f%f",&x0,&eps);
xnew=x0-(x0*x0+x0-1)/(2*x0+1);

```

```

while(fabs(xnew-x0)>eps)
{
x0=xnew;
xnew=x0-(x0*x0+x0-1)/(2*x0+1);
}
printf("the required value is=%f",xnew);
getch();
}

```

5. Regula Falsi Method

```

#include<stdio.h>
#include<conio.h>
#include<math.h>
#define f(x) (x*x+x-1)
void main()
{ int i,n;
float a,b, x[100], eps;
clrscr();
printf("enter the value of n and eps\n");
scanf("%d%f",&n,&eps);
a=0;
b=a+1;
if (f(a)*f(b)>0)
{
a=b;
b=a+1;
}
else
x[0]=a;
x[1]=b;
for(i=1;i<=n;i++)
{
x[i+1]=x[i]-((x[i]-x[i-1])*f(x[i]))/(f(x[i])-f(x[i-1])));
if (fabs(x[i+1]-x[i])<eps)
goto S1;
}
S1:
printf("/n the required value is=%f", x[i-1]);
getch();
}

```

6. Bisection and Secant Method

```

#include<stdio.h>
#include<conio.h>

```

```
#include<math.h>
#define f(x) (x*x*x-9*x+1)
void main()
{ int n=1;
float a,b, x,y1,y2,y3;
clrscr();
printf("enter the interval (a,b) \n");
scanf("%f%f",&a,&b);
y1=f(a);
y2=f(b);
if(y1*y2>0) goto s1;
s2:
x=(a+b)/2;
y3=f(x);
if(y1*y3<0)
b=x;
else
a=x;
n=n+1;
if(fabs(a-b)>.00001) goto s2;
printf("The required root is =%f",x);
s1:
getch();
}
```