

lagrange method

```
x=[1 1.2 1.3 1.5];
y=[0.368 0.301 0.273 0.223];
n=3;
xp=1.4;
sm=0;
tic;
for i=1:n+1
    pr=1
    for j=1:n+1
        if j~=i
            pr=pr*(xp-x(j))/(x(i)-x(j));
        end
    end
    sm=sm + y(i)*pr
end
```

```
pr = 1
sm = 0.0245
pr = 1
sm = -0.1761
pr = 1
sm = 0.1879
pr = 1
sm = 0.2473
```

```
yp=sm;
timetaken = toc
```

```
timetaken = 0.0425
```

newton method

```
f=@(x) (x^3)-5*x + 1;
df=@(x)3*x^2 -5;
xo=2;
tol=1e-6;
n=3;
tic;
for i=1:n
    x1=xo-f(xo)/df(xo);
    if abs(x1-xo)<tol
        end
    xo=x1
end
```

```
xo = 2.1429
xo = 2.1286
xo = 2.1284
```

```
if i==n
end
timetaken = toc
```

```
timetaken = 0.0098
```

fixed point iteration

```

g = @(x) (2^(x)+2)/5;
x0 = 0;
e = 0.000001;
n=6;
% for loop
tic;
for i=1:n;
    x1=g(x0)
    fprintf('x%d =%.4f\n',i,x1)
    if abs(x1 - x0) < e
        break
    end
    x0 = x1
end

```

```

x1 = 0.6000
x1 = 0.6000
x0 = 0.6000
x1 = 0.7031
x2 = 0.7031
x0 = 0.7031
x1 = 0.7256
x3 = 0.7256
x0 = 0.7256
x1 = 0.7307
x4 = 0.7307
x0 = 0.7307
x1 = 0.7319
x5 = 0.7319
x0 = 0.7319
x1 = 0.7322
x6 = 0.7322
x0 = 0.7322

```

```
timetaken = toc
```

```
timetaken = 0.0372
```

secant

```

f=@(x) x-2*cos(x);
xo=1.9;
x1=2;
e=0.0001;
n=5;
tic;
for i=1:n
    %secant method formula
    x2 = (xo*f(x1)-x1*f(xo))/(f(x1)-f(xo))
    fprintf('x%d=% .4f',i,x2)
    xo=x1
    x1=x2
end

```

```

x2 = 1.0087
x1=1.0087
xo = 2
x1 = 1.0087
x2 = 1.0283
x2=1.0283
xo = 1.0087
x1 = 1.0283
x2 = 1.0299
x3=1.0299
xo = 1.0283
x1 = 1.0299
x2 = 1.0299
x4=1.0299
xo = 1.0299
x1 = 1.0299
x2 = 1.0299
x5=1.0299
xo = 1.0299
x1 = 1.0299

```

```
timetaken=toc
```

```
timetaken = 0.0508
```

Trapezoidal

```
% function
f= @(x) x^2*cos(x);
a=0;
b=pi/2;
n=10;
% helps to find the timetaken.
tic
h=(b-a)/n;
% s=summation
s=0.5*(f(a) + f(b));
for i=1:n
    s=s + f(a +i*h)
end
```

```
s = 0.0244
s = 0.1182
s = 0.3161
s = 0.6355
s = 1.0717
s = 1.5938
s = 2.1427
s = 2.6306
s = 2.9433
s = 2.9433
```

```
% computes final answer.
I=h*s
```

```
I = 0.4623
```

```
timetaken=toc
```

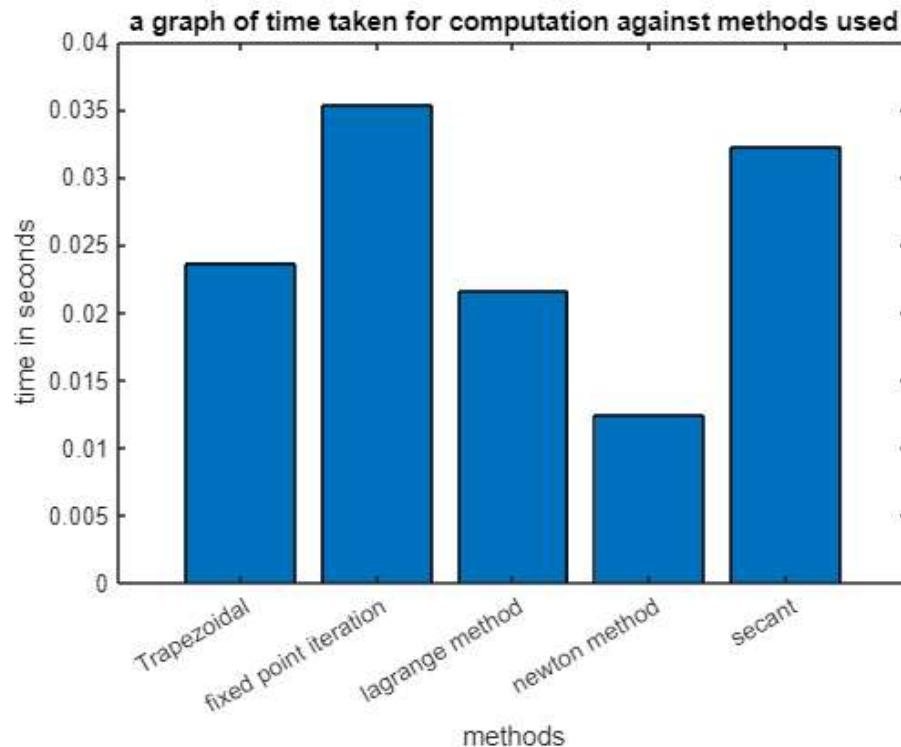
```
timetaken = 0.0258
```

ploting

```

methods={'lagrange method','newton method','fixed point iteration','secant','Trapezoidal'};
timetaken=[0.0216,0.0124,0.0353,0.0323,0.0237];
bar(categorical(methods),timetaken)
hold on;
ylabel('time in seconds')
xlabel('methods')
title('a graph of time taken for computation against methods used')

```



```

% determining the fastest method
[~,idx]=min(timetaken);
fastest_method=methods(idx);
disp(['fastest method:',fastest_method])

```

{'fastest method:'} {'newton method'}