

```

% Newton's Divided Difference (forward) Interpolation
% Input: x=(x1,...,xn)'- x coordination vector
%       y=(y1,...,yn)'- y coordination vector
% Output: D is nx(n+1) upper triangular matrix elements are divided
%         differences and it plots P(x) on interval [a,b].

```

```

function [D]=divdifferen(x,y)
n=length(x);
D=zeros(n,n+1);
D(:,1)=x; D(:,2)=y;
for k=3:n+1;
    for j=1:n+2-k;
        D(j,k)=(D(j+1,k-1)-D(j,k-1))/(D(j+k-2,1)-D(j,1));
    end
end
end

```

---

```

%Interpolating f(x)=exp(x) on [-2,2].

```

```

% Nodes: x1=-2; x2=0; x3=2;

```

```

clear;

```

```

x=-2:2:2;

```

```

y=exp(x);

```

```

D=divdifferen(x,y)

```

```

D =

```

```

-2.0000000000000000    0.135335283236613    0.432332358381694    0.690548922770908
                    0    1.0000000000000000    3.194528049465325    0
    2.0000000000000000    7.389056098930650                    0    0

```

```

a1=D(1,2); a2=D(1,3); a3=D(1,4); %Result: Newton's Polynomial coefficients

```

```

%Plotting y=f(w) and y=P(w)

```

```

w=-2:0.05:2;

```

```

f=exp(w);

```

```

P3=a1+(w+2).*(a2+a3*(w-0))

```

```

plot(w,f,'b',w,P3,'r')

```

```

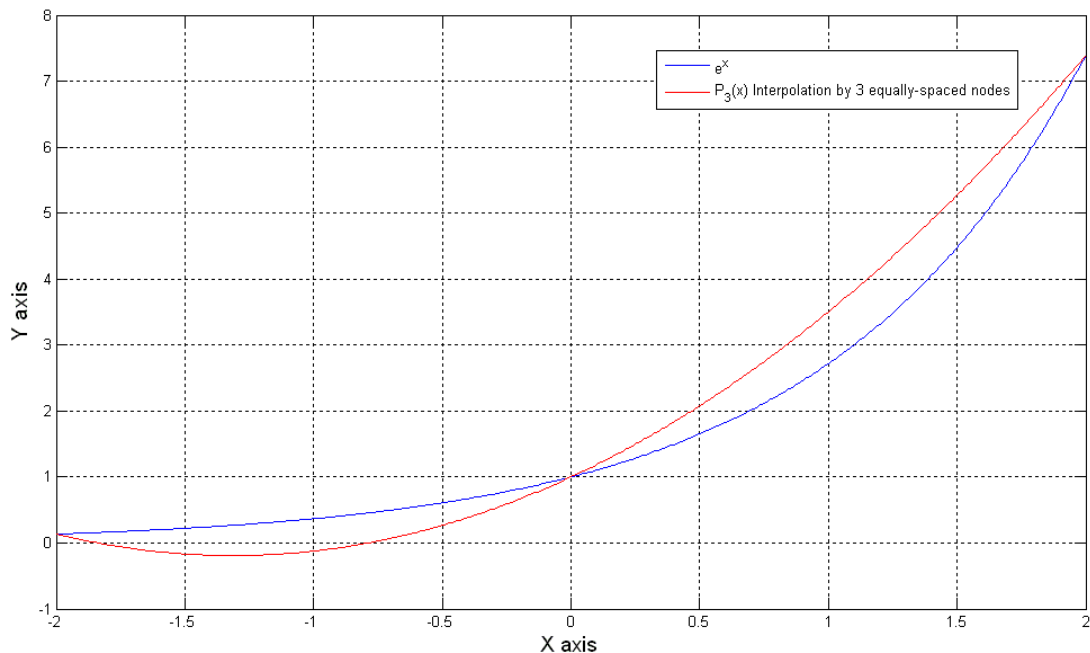
grid

```

```

legend('e^x','P_3(x) Interpolation by 3 equally spaced nodes')

```



```

%Nodes x1=-2; x2=-1; x3=0; x4=1; x5=2
clear;
x=-2:1:2;
y=exp(x);
D=divdifferen(x,y);
a1=D(1,2); a2=D(1,3); a3=D(1,4); a4=D(1,5); a5=D(1,6); %Result: Newton's Polynomial
coefficients
w=-2:0.05:2;
f=exp(w);
P5=a1+(w+2).*(a2+(w+1).*(a3+(w-0).*(a4+a5*(w-1))));
plot(w,f,'b',w,P5,'r')
grid
legend('e^x','P_5(x) Interpolation by 5 equally spaced nodes')

```

