

Apellidos: Serrano Arrese

Nombre: Julia

Incluir códigos empleados, resultados , gráficas y respuestas pedidos. No se darán por válidos los resultados que no se deriven de la secuencia de sentencias incluidas en la solución de cada ejercicio.

Ejercicio 1

INTERPOLACIÓN POLINOMIAL CLÁSICA

%pol.interpolador en 3 nodos equidistantes en intervalo [-1,1]

%nodos interpola

xi = linspace(-1,1,3)';

%valores en los nodos

yi = 1./(1+25.*xi.^2);

%INTERPOLACION POLINOMIAL CLASICA

H=[xi.^0 xi.^1 xi.^2];

b=yi;

c = H\b

xx = -1:0.001:1;

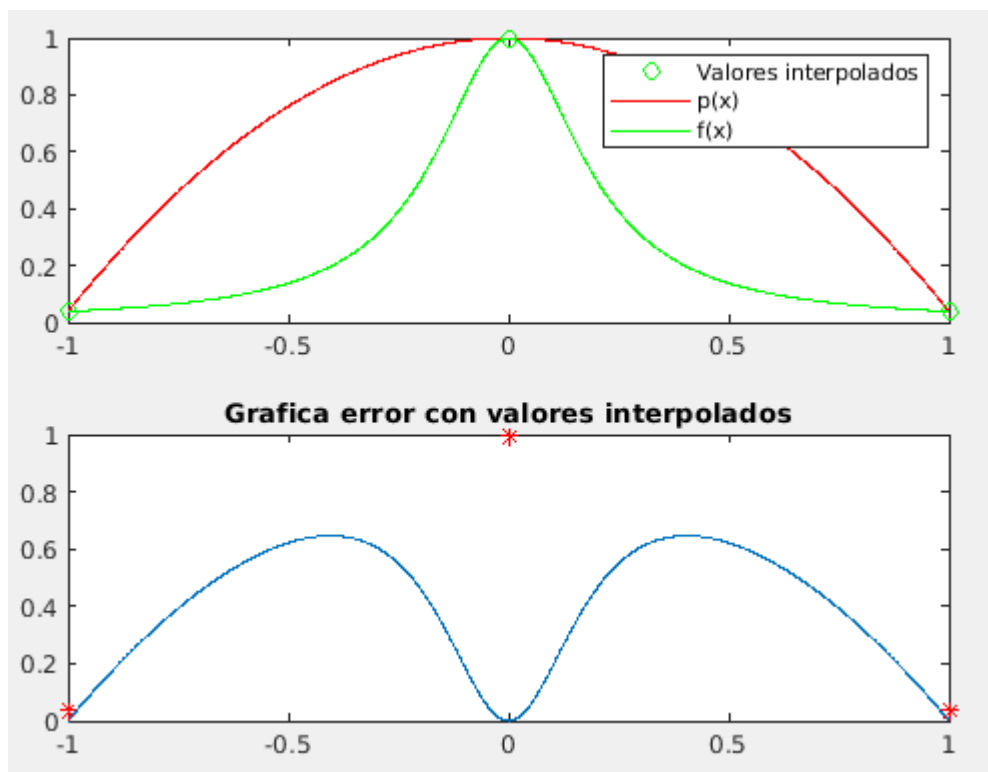
px = c(1)*xx.^0+c(2)*xx.^1+c(3)*xx.^2;

fx = 1./(1+25.*xx.^2);

subplot(2,1,1)

```
plot(xi,yi,'go',xx,px,'r',xx,fx,'g')
legend('Valores interpolados','p(x)','f(x)')
```

```
%error interpolacion en xx
error_interp = abs(fx-px);
max_error = max(error_interp);
subplot(2,1,2)
plot(xx,error_interp,xi,yi,'r*')
title('Grafica error con valores interpolados')
```



INTERPOLACIÓN POR FÓRMULA DE NEWTON

```
clear;
clc;
%pol.interpolador en 3 nodos equidistantes en intervalo [-1,1]
```

```
%nodos interpola
```

```
xi = linspace(-1,1,3);
```

```
%valores en los nodos
```

```
yi = 1./(1+25.*xi.^2);
```

```
%Evaluacion del polinomio
```

```
xx = -1:0.01:1;
```

```
fx = 1./(1+25.*xx.^2);
```

```
px = polNewton(xi',yi',xx);
```

```
%grafica
```

```
%polinomio obtenido:'r'
```

```
%funcion 'g'
```

```
%valores interpolados 'go'
```

```
subplot(2,1,1)
```

```
plot(xi,yi,'ro',xx,px,'r',xx,fx,'g')
```

```
%error interpolacion en xx
```

```
error_interp = abs(fx-px);
```

```
max_error = max(error_interp);
```

```
max_error =
```

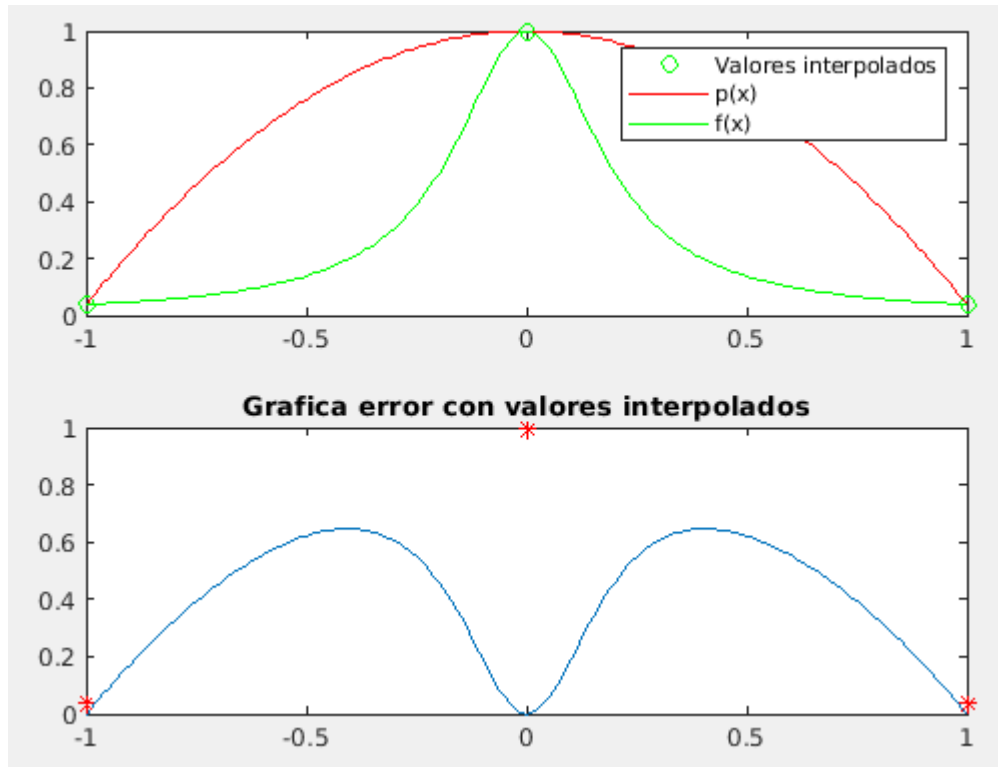
```
0.6462
```

```
%grafica error
```

```
subplot(2,1,2)
```

```
plot(xx,error_interp,xi,yi,'r*')
```

```
title('Grafica error con valores interpolados')
```



2.

```
function c = coef_interp(xi,yi)
```

```
%Implementa el calculo de los coef. polinomio interpolacion en datos
```

```
%xi,yi
```

```
%Arg.entrada: xi,yi (vectores columna)
```

```
%Arg. salida: c vector de coeficientes del polinomio
```

```
l=length(xi);
```

```
D=zeros(l);
```

```
D(:,1)=yi;
```

```
for k=2:l
```

```
for j=1:l-k+1
```

```
dif=D(j,k-1)-D(j+1,k-1);
```

```
dx=xi(j)-xi(j+k-1);
```

```
D(j,k)=dif/dx;
```

```
end
```

```
end
```

```
% Vector con coeficientes F. Newton
```

```
c=D(1,:);
```

```
end
```

¿Qué grado tendrá el polinomio y cuál será la dimensión de la matriz H de coeficientes del sistema?

Grado será la longitud de xi - 1. H tendrá dimensión lxl

3.

```
n_nodos = [3 5 9 17];
```

```
l = length(n_nodos) ;
```

```
xx = -1:0.01:1;
```

```
pp = zeros(4*length(xx));
```

```
for i = 1:l
```

```
    %distancia entre nodos consecutivos
```

```
    h = 1/2^(n_nodos(i)-1);
```

```
    %nodos donde se interpola
```

```
    xi = [-1:h:1]';
```

```
    %valores de f(x) en xi
```

```
    yi = 1./(1+25.*xi.^2);
```

```
    %coeficientes
```

```
    c = 1./(1+25.*xi.^2);
```

```
end
```

```
c_inv = c(end:-1:1);
```

```
pp = polyval(c_inv,xx)
```

```
error = zeros(4*length(xx));
```