```
close all
%Function and variable declarations
nodes = [];
en1 = [];
en2 = [];
u1 = [];
u2 = [];
n = [100; 150; 200; 250; 300; 350; 400; 450; 500; 550; 600; 650; 700; 750; 800; 850; 900;↙
950; 1000];
%Choose the values we wish to interpolate at
interpX = [-.96, -.71, -.51, -.22, -.12, -.08, 0, .08, .12, .22, .51, .71, .96];
for i=1:length(n)
    xi = [];
    %Compute the equally-spaced nodes
    xi = -1:(2*i)/n(i, 1):1;
    xi=xi';
    %Compute the barryweights for xi
    bw = baryweights(xi);
    %Compute the barryweights for the Chebyshev nodes
    Cbw = chebs(length(1:n(i))+1);
    %Compute the Chebyshev nodes
    Cxi = chebXi(1:n(i));
    %Calculate the y values of the function at each x in xi
    y1 = [];
    y2 = [];
    for j=1:length(xi)
        y1(j,1) = f1(xi(j,1));
        y2(j,1) = f2(xi(j,1));
    end
    %Calculate the y values of the function at each x in Cxi
    Cy1 = [];
    Cy2 = [];
    Cy3 = [];
    for j=1:length(Cxi)
        Cy1(j,1) = f1(Cxi(j,1));
        Cy2(j,1) = f2(Cxi(j,1));
        Cy3(j,1) = f3(Cxi(j,1));
    end
    %bary interp for xi
    u1 = baryinterp(xi, bw, y1, interpX);
    u2 = baryinterp(xi, bw, y2, interpX);
    %bary interp for Cxi
    Cu1 = baryinterp(Cxi, Cbw, Cy1, interpX);
    Cu2 = baryinterp(Cxi, Cbw, Cy2, interpX);
    Cu3 = baryinterp(Cxi, Cbw, Cy3, interpX);
    %Compute the error of the interpolating polynomial for u1, u2
    eVec1 = [];
    eVec2 = [];
    en1(i) = 0;
    en2(i) = 0;
    for j=1:length(interpX)
```

```
        x = interpX(j);
        e1 = abs(u1(j) - f1(x));
        e2 = abs(u2(j) - f2(x));
        if e1 > en1(i)
            en1(i) = e1;
        end
        if e2 > en2(i)
            en2(i) = e2;
        end
    end
    %Compute the error of the interpolating polynomial for Cu1, Cu2, Cu3
    Cen1(i) = 0;
    Cen2(i) = 0;
    Cen3(i) = 0;
    for j=1:length(interpX)
        x = interpX(j);
        Ce1 = abs(Cu1(j) - f1(x));
        Ce2 = abs(Cu2(j) - f2(x));
        Ce3 = abs(Cu3(j) - f3(x));
        if Ce1 > Cen1(i)
            Cen1(i) = Ce1;
        end
        if Ce2 > Cen2(i)
            Cen2(i) = Ce2;
        end
        if Ce3 > Cen3(i)
            Cen3(i) = Ce3;
        end
    end
end

%Plot log10(en) vs n for f1
hold on;
grid on;
axis on;
ylabel("log(En)");
xlabel("N");
title("log(En) vs f(x) = 1/(5-4x) with even nodes");
plot(n, abs(log(en1)), "r*");
hold off
figure;

%Plot log10(en) vs n for f2
hold on;
grid on;
axis on;
ylabel("log(En)");
xlabel("N");
title("log(En) vs f(x) = 1/(1+16*(x^2)) with even nodes");
plot(n, log(en2), "r*");
hold off
```

```
figure;

%Plot log10(en) vs n for f1 C
hold on;
grid on;
axis on;
ylabel("log(En)");
xlabel("N");
title("log(En) vs f(x) = 1/(5-4x) with Chebyshev nodes");
plot(n, log(Cen1), "r*");
hold off
figure;

%Plot log10(en) vs n for f2 C
hold on;
grid on;
axis on;
ylabel("log(En)");
xlabel("N");
title("log(En) vs f(x) = 1/(1+16*(x^2)) with Chebyshev nodes");
plot(n, log(Cen2), "r*");
hold off
figure;

%Plot log10(en) vs n for f3 C
hold on;
grid on;
axis on;
ylabel("log(En)");
xlabel("N");
title("log(En) vs f(x) = cos(((10)^4)*x) with Chebyshev nodes");
plot(n, log(Cen3), "r*");
hold off

%Find the smallest value of n (to within ±10) such that en ≤ 10^−5
N = 100000;
while true
    N = N + 10;
    %Compute the barryweights for the Chebyshev nodes
    Cbw = chebs(length(1:N)+1);
    %Compute the Chebyshev nodes
    Cxi = chebXi(1:N);
    %Calculate the y values of the function at each x in Cxi
    Cy = [];
    for j=1:length(Cxi)
        Cy(j,1) = f3(Cxi(j,1));
    end
    Cu = baryinterp(Cxi, Cbw, Cy, interpX);
    %Compute the error of the interpolating polynomial
    Cen = 0;
    for j=1:length(interpX)
```

```matlab
            x = interpX(j);
            Ce = abs(Cu(j) - f3(x));
            if Ce > Cen
                Cen = Ce;
            end
        end
        if Cen <= 1e-5
            disp(N)
            break
        end
end

%Functions
function y = f1(x)
    y=1/(5-4*x);
end
function y = f2(x)
    y=1/(1+16*(x^2));
end
function y = f3(x)
    y=cos(((10)^4)*x);
end
function w = chebs(n)
    w = zeros(n, 1);
    w(1, 1) = 1/2;
    w(n) = .5*((-1)^n);
    for i=2:n
        w(i) = (-1)^(i-2);
    end
end
function xi = chebXi(x)
    xi = zeros(length(x),1);
    for i=1:length(x)+1
        xi(i) = cos(((i-1)*pi)/length(x));
    end
    xi = flip(xi);
end
```