1.4 Run Program 1 to $N = 2^{16}$ instead of 2^{12}. What happens to the plot of the error vs. N? Why? Use the MATLAB commands tic and toc to generate a plot of approximately how the computation time depends on N. Is the dependence linear, quadratic, or cubic?

Solution.
We can see from these two images that the plot of the error starts to behave differently at the end. This is due to the computations near the limit of the computational precision of the program. We can see that the error starts to go up as N gets closer to this precision after the point of $N = 2^{12}$.

212.PNG

Figure 1: $N = 2^{12}$

216.PNG

Figure 2: $N = 2^{16}$

Now, as we see by the time graph, the dependance appears to be more linear. Even with the cluster of time on the left, the data seems to be well approximated by a linear function.

tictoc4.PNG

Figure 3: Time vs N

```
1 close all
2 clc
3 % p1.m - convergence of fourth-order finite differences
4
5 % For various N, set up grid in [-pi,pi] and function u(x):
6   Nvec= 2.^(3:16);
7   clf, subplot('position',[.1 .4 .8 .5])
8   for N = Nvec
9       tic
10    h = 2*pi/N; x = -pi + (1:N)'*h;
11    u = exp(sin(x)); uprime = cos(x).*u;
12
13    % Construct sparse fourth-order differentiation matrix:
14    e = ones(N,1);
15    D =   sparse(1:N,[2:N 1],2*e/3,N,N)...
16        - sparse(1:N,[3:N 1 2],e/12,N,N);
17    D = (D-D')/h;
18
19    % Plot max(abs(D*u-uprime)):
20    error = norm(D*u-uprime,inf);
21    loglog(N,error,'.','markersize',15),
22    hold on;
23    T=toc;
24    %scatter(N,T);
25    %hold on;
26
27  end
28  grid on, xlabel N, ylabel error
29  title('Convergence of fourth-order finite differences')
30  semilogy(Nvec,Nvec.^(-4),'--')
31  text(105,5e-8,'N^{-4}','fontsize',18)
32  %grid on, xlabel N, ylabel Time
33  %title('Computation time for Different N Values')
34
35 % Uncomment out the last two lines in the for loop and the last ...
     two lines
36 % above while commenting out the loglog line and lines 28-31 to ...
     get the
37 % code used to generate the time graph.
```

1.5 Run Programs 1 and 2 with $e^{\sin(x)}$ replaced by (a) $e^{\sin^2(x)}$ and (b) $e^{\sin(x)|\sin(x)|}$ and with uprime adjusted appropriately. What rates of convergence do you observe?

Solution.
a. The rates of convergence seem to be roughly the same as that of the original $e^{\sin(x)}$ function. They might be a little slower, though the convergence is still very much present and similar to that of the original.

151a.PNG

152a.PNG

b. The rates of convergence for $e^{\sin(x)|\sin(x)|}$ are drastically different than that of part a and the original. That is to say, the error tends upward instaed of downward as the iterations increase. This is due to discontinuities in the derivative of the function.

151b.PNG

1 52b.PNG

```
clear
close all
clc
% p1.m - convergence of fourth-order finite differences

% For various N, set up grid in [-pi,pi] and function u(x):
  Nvec = 2.^(3:12);
  clf, subplot('position',[.1 .4 .8 .5])
  for N = Nvec
    h = 2*pi/N; x = -pi + (1:N)'*h;
    u = exp(sin(x).^2); uprime = 2.*sin(x).*cos(x).*u;

    % Construct sparse fourth-order differentiation matrix:
    e = ones(N,1);
    D =   sparse(1:N,[2:N 1],2*e/3,N,N)...
        - sparse(1:N,[3:N 1 2],e/12,N,N);
    D = (D-D')/h;

    % Plot max(abs(D*u-uprime)):
    error = norm(D*u-uprime,inf);
    loglog(N,error,'.','markersize',15), hold on
  end
  grid on, xlabel N, ylabel error
  title('Convergence of fourth-order finite differences')
  semilogy(Nvec,Nvec.^(-4),'--')
  text(105,5e-8,'N^{-4}','fontsize',18)
```

```
clear
close all
clc
% p2.m - convergence of periodic spectral method (compare p1.m)

% For various N (even), set up grid as before:
  clf, subplot('position',[.1 .4 .8 .5])
  for N = 2:2:100;
    h = 2*pi/N;
    x = -pi + (1:N)'*h;
    u = exp(sin(x).^2); uprime = 2.*sin(x).*cos(x).*u;

    % Construct spectral differentiation matrix:
    column = [0 .5*(-1).^(1:N-1).*cot((1:N-1)*h/2)];
    D = toeplitz(column,column([1 N:-1:2]));

    % Plot max(abs(D*u-uprime)):
    error = norm(D*u-uprime,inf);
    loglog(N,error,'.','markersize',15), hold on
  end
  grid on, xlabel N, ylabel error
  title('Convergence of spectral differentiation')
```

```
clear
close all
clc
% p1.m - convergence of fourth-order finite differences

% For various N, set up grid in [-pi,pi] and function u(x):
  Nvec = 2.^(3:12);
  clf, subplot('position',[.1 .4 .8 .5])
  for N = Nvec
    h = 2*pi/N; x = -pi + (1:N)'*h;
    u = exp(sin(x).*abs(sin(x))); uprime = ...
        (sin(x).*sin(2.*x)/(abs(sin(x)))).*u;

    % Construct sparse fourth-order differentiation matrix:
    e = ones(N,1);
    D =   sparse(1:N,[2:N 1],2*e/3,N,N)...
        - sparse(1:N,[3:N 1 2],e/12,N,N);
    D = (D-D')/h;

    % Plot max(abs(D*u-uprime)):
    error = norm(D*u-uprime,inf);
    loglog(N,error,'.','markersize',15), hold on
  end
  grid on, xlabel N, ylabel error
  title('Convergence of fourth-order finite differences')
  semilogy(Nvec,Nvec.^(-4),'--')
  text(105,5e-8,'N^{-4}','fontsize',18)
```

```
clear
close all
clc
% p2.m - convergence of periodic spectral method (compare p1.m)

% For various N (even), set up grid as before:
  clf, subplot('position',[.1 .4 .8 .5])
  for N = 2:2:100;
    h = 2*pi/N;
    x = -pi + (1:N)'*h;
    u = exp(sin(x).*abs(sin(x))); uprime = ...
        (sin(x).*sin(2.*x)/(abs(sin(x)))).*u;

    % Construct spectral differentiation matrix:
    column = [0 .5*(-1).^(1:N-1).*cot((1:N-1)*h/2)];
    D = toeplitz(column,column([1 N:-1:2]));

    % Plot max(abs(D*u-uprime)):
    error = norm(D*u-uprime,inf);
    loglog(N,error,'.','markersize',15), hold on
  end
  grid on, xlabel N, ylabel error
  title('Convergence of spectral differentiation')
```

1.6 By manipulating Taylor series, determine the constant C for an error expansion of (1.3) of the form $w_j - u'(x_j) \sim Ch^4 u^{(5)}(x_j)$, where $u^{(5)}$ denotes the fifth derivative. Based on this value of C and on the formula for $u^{(5)}(x)$ with $u(x) = e^{\sin(x)}$ determine the leading term in the expansion for $w_j - u'(x_j)$ for $u(x) = e^{\sin(x)}$ (You will have to find $\max_{x\in[-\pi,\pi]} |u^{(5)}(x)|$ numerically.) Modify Program 1 so that it plots the dashed line corresponding to this leading term rather than just N^{-4}. This adjusted dashed line should fit the data almost perfectly. Plot the difference between the two on a log-log scale and verify that it shrinks at the rate $\mathcal{O}(h^6)$.

Solution.
We are given the differential matrix

$$D = \frac{1}{h}\begin{pmatrix} \ddots & \ddots & \cdots & \frac{1}{12} & -\frac{2}{3} \\ \ddots & \ddots & -\frac{1}{12} & \ddots & \frac{1}{12} \\ \ddots & \ddots & \frac{2}{3} & \ddots & \ddots \\ \ddots & \ddots & 0 & \ddots & \ddots \\ \ddots & \ddots & -\frac{2}{3} & \ddots & \ddots \\ -\frac{1}{12} & \ddots & \frac{1}{12} & \ddots & \ddots \\ \frac{2}{3} & -\frac{1}{12} & \cdots & \ddots & \ddots \end{pmatrix}$$

So we can, from 1.3, write the equation

$$w_j = \frac{1}{h}(-\frac{1}{12}u_{j-2} + \frac{2}{3}u_{j-1} - \frac{2}{3}u_{j+1} + \frac{1}{12}u_{j+2})$$

Therefore, by expanding each non u_j term about u_j, we get

$$\begin{aligned} w_j = \frac{1}{h}(\frac{1}{12}(-u + 2hu' - 2h^2u'' + \frac{8}{6}h^3u''' - \frac{2}{3}h^4u'''' + \frac{1}{45}h^5u''''') \\ +\frac{2}{3}(u - hu' + \frac{1}{2}h^2u'' - \frac{1}{6}h^3u''' + \frac{1}{24}h^4u'''' - \frac{1}{120}h^5u''''') \\ +\frac{2}{3}(-u - hu' - \frac{1}{2}h^2u'' - \frac{1}{6}h^3u''' - \frac{1}{24}h^4u'''' - \frac{1}{120}h^5u''''') \\ +\frac{1}{12}(u + 2hu' + 2h^2u'' + \frac{8}{6}h^3u''' + \frac{2}{3}h^4u'''' + \frac{1}{45}h^5u''''')) \\ = \frac{1}{h}(-hu' + \frac{1}{30}h^5u''''') = -u' + \frac{1}{30}h^4u''''' \end{aligned}$$

where u and its derivatives denote $u(x_j)$. Hence, we get a C value of $C = \frac{1}{30}$. Now, taking the max value of $u^{(5)}$ on $[-\pi, \pi]$, we get $\max|u^{(5)}| = 24.811$. Therefore, using

16.PNG

the above equation and C value that we found, we get the following graph where the approximation is almost perfectly in line with the dotted line.

```
clear
close all
clc
% p1.m - convergence of fourth-order finite differences

% For various N, set up grid in [-pi,pi] and function u(x):
  Nvec = 2.^(3:12);
  Mvec = zeros(length(Nvec),1);
  counter = 1;
  clf, subplot('position',[.1 .4 .8 .5])
  for N = Nvec
    h = 2*pi/N; x = -pi + (1:N)'*h;
    u = exp(sin(x)); uprime = cos(x).*u;

    % Construct sparse fourth-order differentiation matrix:
    e = ones(N,1);
    D =   sparse(1:N,[2:N 1],2*e/3,N,N)...
        - sparse(1:N,[3:N 1 2],e/12,N,N);
    D = (D-D')/h;

    % Plot max(abs(D*u-uprime)):
    error = norm(D*u-uprime,inf);
    Mvec(counter)=(1/30).*h^4.*24.811;
    counter=counter+1;
    loglog(N,error,'.','markersize',15), hold on
  end
  grid on, xlabel N, ylabel error
  title('Convergence of fourth-order finite differences')
  semilogy(Nvec,Mvec,'--')
```

2.6 We obtained the entries of (1.4) by differentiating the sinc function. Derive the same result by calculating the inverse semidiscrete Fourier transform of $ik\hat{\delta}(k)$.

Solution.
We need to calculate the inverse semidiscrete Fourier transform of the above expression by plugging it into

$$\frac{1}{2\pi}\int_{-\frac{\pi}{h}}^{\frac{\pi}{h}} e^{ikx}\hat{v}(k)dk$$

where $\hat{\delta}(k)$ is given by h. So, we get

$$\frac{1}{2\pi}\int_{-\frac{\pi}{h}}^{\frac{\pi}{h}} e^{ikx}ik\hat{\delta}(k)dk = \frac{2\pi x\cos(\frac{\pi x}{h}) - 2h\sin(\frac{\pi x}{h})}{2\pi x^2}$$

Now, when we differentiate the sinc function $sinc(\frac{\pi x}{h})$, we arrive at

$$\frac{2\pi x\cos(\frac{\pi x}{h}) - 2h\sin(\frac{\pi x}{h})}{2\pi x^2}$$

Thus, computing the inverse semidiscrete Fourier transform on $ik\hat{\delta}(k)$ gives the same results as differentiating $sinc(\frac{\pi x}{h})$.

2.7 Modify Program 3 to determine the maximum error over $\mathbb{R}$ in the sinc function interpolants of the square wave and the hat function, and to produce a log-log plot of these two error maxima as functions of h. (Good choices for h are 2^{-3}, 2^{-4}, 2^{-5} , 2^{-6}.) What convergence rates do you observe as $h \to 0$?

Solution.
Here, we have the graphs of the functions with increasing h values, and we can clearly see that the error for the computations decreases as $h \to 0$. It also seems to settle after $h = 2^{-5}$.

273.PNG

274.PNG

275.PNG

I was not sure how to get the loglog plot to work.

276.PNG

```
% Stephanie Klumpe
% Problem 2.7 edited code

clear
close all
clc

hval = [2^-3 2^-4 2^-5 2^-6];                 %% h-value vector
maxe = zeros(2,4);                           %% Max error matrix for ...
    both fxns
counter = 1;                                  %% Indexing counter

for h = hval

  xmax = 10; clf
  x = -xmax:h:xmax;                          % computational grid %% ...
      Estimate?
  xx = -xmax-h/20:h/10:xmax+h/20;            % plotting grid %% Actual?

  for plt = 1:2
    switch plt
        case 1, v = (abs(x)≤3);              % square wave
                vv = (abs(xx)≤3);

        case 2, v = max(0,1-abs(x)/3);        % hat function
                vv = max(0,1-abs(xx)/3);
    end

    p = zeros(size(xx));
    for i = 1:length(x)
      p = p + v(i)*sin(pi*(xx-x(i))/h)./(pi*(xx-x(i))/h);
    end

    maxe(plt,counter) = norm(p-vv,Inf);        %% Max error entries
  end

 counter = counter+1;
end

subplot(2, 1, 1);
loglog(hval,maxe(1,:))                        %% Plot of h vs error of sw
subplot(2, 1, 2);
loglog(hval,maxe(2,:))                        %% Plot of h vs error of hf
```

3.5 Using the commands tic and toc, study the time taken by MATLAB (which is far from optimal in this regard) to compute an N-point FFT, as a function of N. If $N = 2^k$ for $k = 0, 1, ..., 15$, for example, what is the dependence of the time on N? What if $N = 500, 501, ..., 519, 520$? From a plot of the latter results, can you spot the prime numbers in this range? (Hints. The commands isprime and factor may be useful. To get good tic/toc data it may help to compute each FFT 10 or 100 times in a loop.)

Solution.
For N in the first case, we have that the time for the FFT on $2\sin(3x + 2)$ decreases repidly and then slowly increases. For N in the second case, we have that the FFT on the same function countinuously decreses in terms of the computational time required.

```
clear
close all
clc

for k=[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]
    N=2.^k;
    tic
    h = 2*pi/N; x = -pi + (1:N)'*h;
    u=2.*sin(3.*x+2);
    uhat=fft(u,N);
    toc
end
fprintf('\n')
for N = 500:1:520
    tic
    h = 2*pi/N; x = -pi + (1:N)'*h;
    u=2.*sin(3.*x+2);
    uhat=fft(u,N);
    toc
end
```

3.6 We have seen that a discrete function v can be spectrally differentiated by means of two complex FFTs (one forward, one inverse). Explain how two distinct discrete functions v and w can be spectrally differentiated at once by the same two complex FFTs, provided that v and w are real.

Solution.
Let v and w be two real valued discrete functions. Then define the function $u \equiv v + iw$. Then, we have that u is a complex function and we know that we can spectrally differentiate u by the use of the FFT and its inverse. So, by applying the FFT to u, we get $\hat{u} = \hat{v} + i\hat{w}$. We can then spectrally differentiate u and by consequence spectrally differentiate v and w. Succeding this, we then apply the inverse FFT to $\hat{u}'$ and obtain u' and therefore v' and w'.

4.3 (a) Determine the Fourier transform of $u(x) = \frac{1}{(1+x^2)}$. (Use a complex contour integral if you know how; otherwise, copy the result from (4.3).) (b) Determine $\hat{v}(k)$, where v is the discretization of u on the grid $h\mathbb{Z}$. (Hint. Calculating $\hat{v}(k)$ from the definition (2.3) is very difficult.) (c) How fast does $\hat{v}(k)$ approach $\hat{u}(k)$ as $h \to 0$? (d) Does this result match the predictions of Theorem 3?

Solution.
Using reslut 4.3 and taking $\sigma = 1$, we have $\hat{u}(k) = \pi e^{-|k|}$ as the Fourier transform of $u(x)$. Now, to determine the semidiscrete Fourier transform of u on the grid $h\mathbb{Z}$, let $v = u(x_j)$ where $x_j \in h\mathbb{Z}$. Using Theorem 2, we have that

$$\hat{v}(k) = \sum_{j=-\infty}^{\infty} \hat{u}(k + \frac{2\pi j}{h}) = \sum_{j=-\infty}^{\infty} \pi e^{-|k+\frac{2\pi j}{h}|}$$

So, leaving $\hat{v}$ in this form, and by the fact that $u(x)$ has infinitely many continuous derivatives, we see that $|\hat{u} - \hat{v}| = \mathcal{O}(h^m)$ for all $m \in \mathbb{Z}^+$. That is the order of the difference of the two transformations is better than any power of h as $h \to 0$. This also fits with the predictions of Theorem 3, which is expected.