Departamento de Ingeniería Eléctrica y Computadoras
Universidad de Puerto Rico - Mayagüez

Procesamiento Digital de Señales
Tarea de Práctica Individual No. 1
Tratamiento Básico de Señales

Solución
Primera Tarea

Por
Prof. Domingo Rodríguez
Departamento de Ingeniería Eléctrica y Computadoras
Universidad de Puerto Rico en Mayagüez
domingo@ece.uprm.edu

13 de Noviembre de 2003

Tarea de Práctica Individual No. 1 – Tratamiento Básico de Señales
Fecha de entrega: 27 de enero de 2005 antes de las 11:59PM

Introducción:
Esta primera tarea tiene como propósito principal introducir al estudiante a conceptos básicos de tratamiento de señales. Cada estudiante procederá a estudiar los problemas asignados en este documento y proveer sus "**propias soluciones originales**" agregadas al mismo documento. Los archivos de MATLAB que aparecen el documento para ser modificados provienen del libro de laboratorio que acompaña al libro de texto de la clase. Las gráficas generadas en este documento provienen también de los archivos de MATLAB.

Para aquellos estudiantes que estén interesados, los programas, e información adicional, pueden ser obtenidos directamente del servidor de la universidad del Prof. Mitra a través de uso de FTP ("File Transfer Protocol"). Sin embargo, este documento contiene toda la información necesaria para efectuar la tarea apropiadamente. La siguiente información permite acceder el servidor a través de FTP:

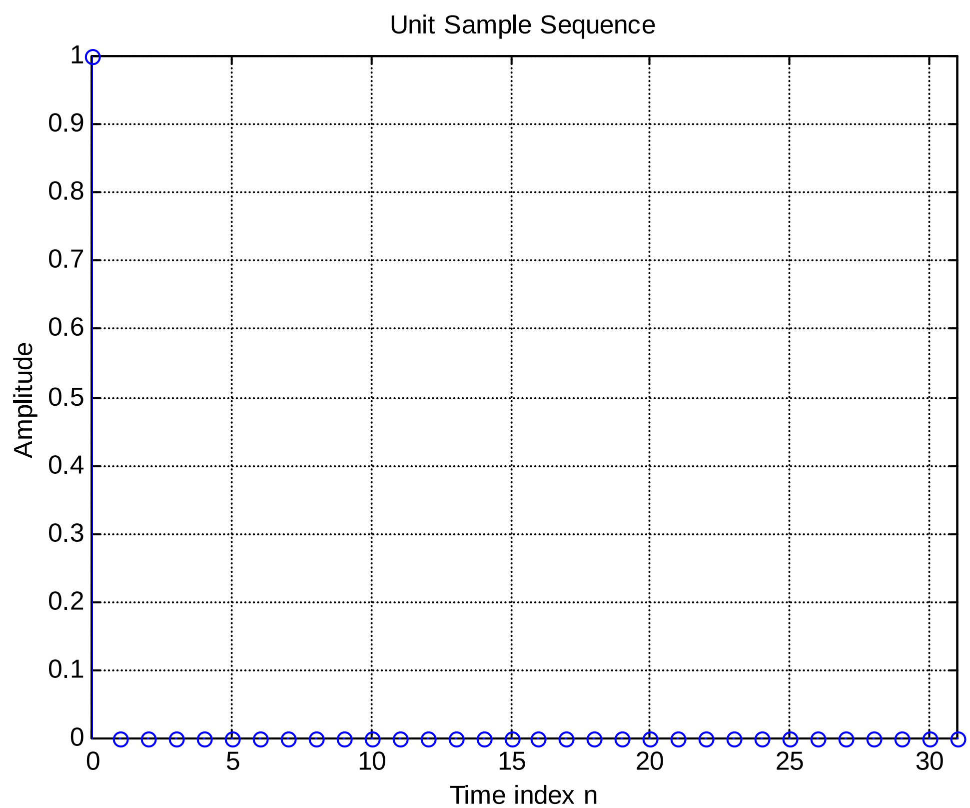
Host: **ftp://iplserv.ece.ucsb.edu**
User ID: **anonymous**
Password: **dirección de e-mail del estudiante**
Directory: **/pub/mitra/Labs/pc**

Este documento debe ser entregado con las siguientes modificaciones e indicaciones:
1. Enviar copia del documento vía e-mail utilizando las reglas facilitadas
2. Colocar nombre, número de matrícula y fecha de entrega en la primera página
3. Dejar la parte de la **Introducción** sin ser modificada
4. Modificar cada programa en MATLAB según las indicaciones dadas
5. Sustituir las gráficas dadas por las generadas de los programas modificados
6. Proveer los desarrollos matemáticos que acompañan las demostraciones pedidas

Problema No. P1.1: Señal de impulso unitario

Modificar el siguiente programa en MATLAB para graficar la señal $\delta[n], \quad n \in Z_{32}$, la cual es el impulso unitario, y sustituir la siguiente gráfica por la gráfica resultante del programa modificado. Note que la gráfica debe comenzar con el índice de 0 y terminar con el índice de 31.

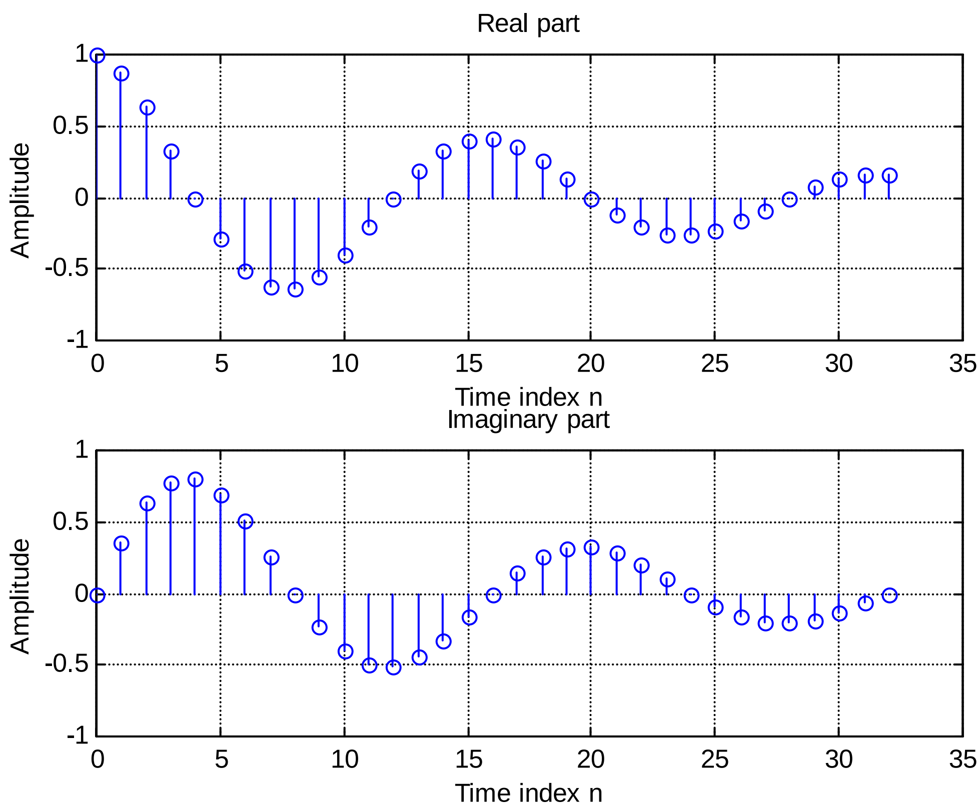
```
% Program P1_1 - Modified by Prof. D. Rodriguez - Fall Semester 2003
% Generation of a Unit Sample Sequence
clf;
% Generate a vector from 0 to 31
n = 0:31;
% Generate the unit sample sequence
u = [1 zeros(1,31)];
% Plot the unit sample sequence
stem(n,u);
xlabel('Time index n');ylabel('Amplitude');
title('Unit Sample Sequence');
axis([0 31 0 1]);
grid
```



Problema No. P1.2: Secuencia exponencial compleja

Modificar el siguiente programa para graficar la señal $x[n] = A^{-\frac{n}{4}} e^{+j\frac{2\pi}{32}(2)n}$, $n \in Z_{32}$, la cual es el producto dos señales particulares: una señal compleja periódica, con período fundamental de longitud 8, por una señal exponencial real atenuante, con parámetro variable $1 \le A \le 4$. Cada estudiante deberá escoger un "**valor único**" de $1 \le A \le 4$ y sustituir la siguiente gráfica por la gráfica resultante del programa modificado. Note que la señal $x[n] = A^{-\frac{n}{4}} e^{+j\frac{2\pi}{32}(2)n}$, $n \in Z_{32}$ es una señal compleja y se debe graficar su parte real y su parte imaginaria.

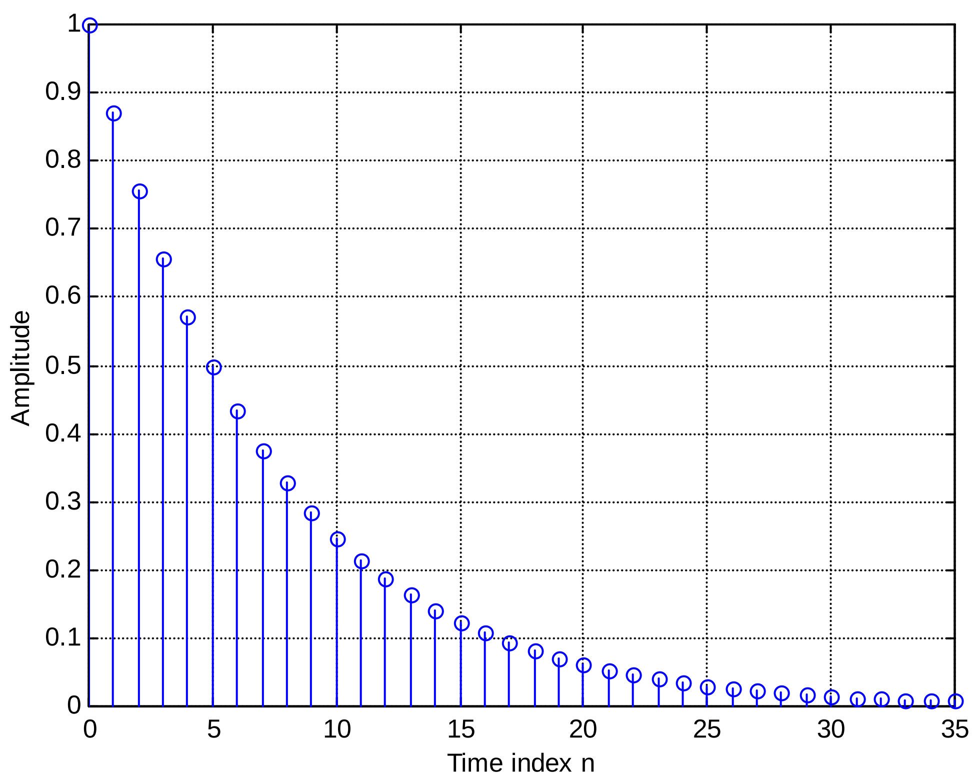
```
% Program P1_2 - Modified by Prof. Domingo Rodriguez - Fall 2003
% Generation of a complex exponential sequence
clf;
f = (2/32); A = 1.25;   n = 0:32;  phase = 0;
arg = +j*(2*pi*f*n - phase);
x = (A.^-(n/4)).*exp(arg);
subplot(2,1,1);
stem(n,real(x));
xlabel('Time index n');ylabel('Amplitude');
title('Real part');
grid
subplot(2,1,2);
stem(n,imag(x));
xlabel('Time index n');ylabel('Amplitude');
title('Imaginary part');
grid
```



Problema No. P1.3: Señal discreta (secuencia) exponencial real

Modificar el siguiente programa para graficar la señal $x[n] = e^{(-1)\frac{2\pi}{36}(0.8)n}$, $n \in Z_{36}$, la cual es una señal real exponencial. Note que la señal resultante decae al aumentar el tiempo y tiene como asíntota el eje de la abscisa. El comportamiento de la señal resultante es diferente al comportamiento de la señal graficada más abajo y producida por programa dado para ser modificado.

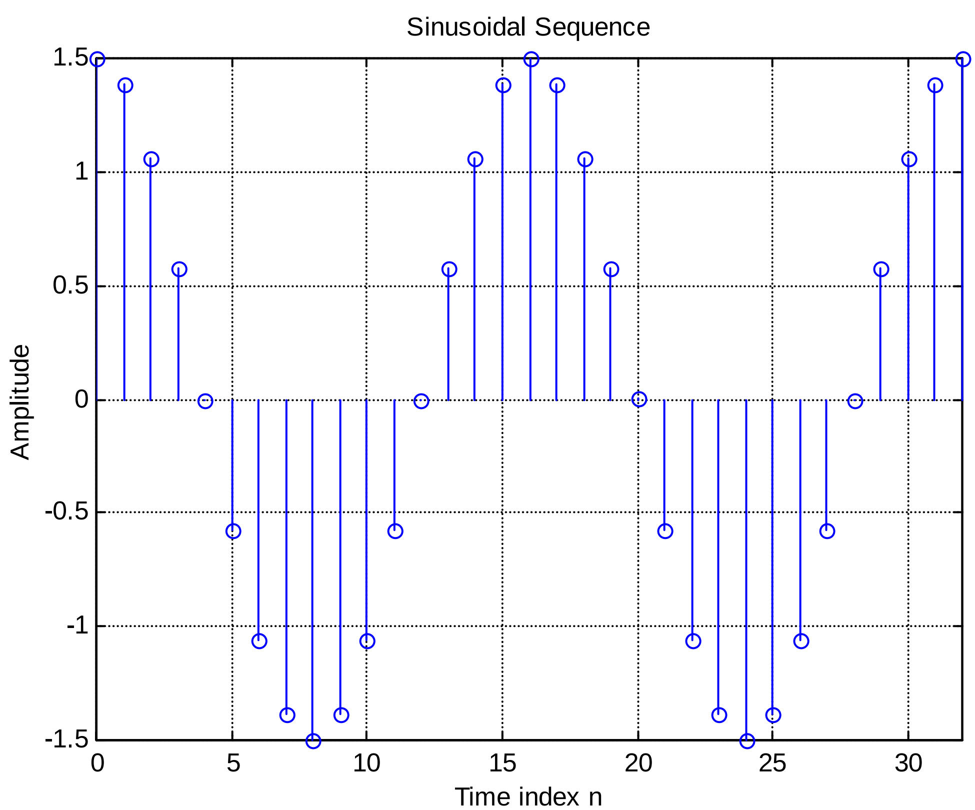
```
% Program P1_3 - Modified by Prof. Domingo Rodriguez - Fall 2003
% Generation of a real exponential sequence
clf;
n = 0:35; a = 1.2; K = 0.2;
x = exp(-(2*pi/36)*0.8*n);
stem(n,x);
xlabel('Time index n');ylabel('Amplitude');
grid
```



Problema No. P1.4: Señal discreta (secuencia) sinusoidal

Modificar el siguiente programa para graficar la señal $x[n] = 1.5\cos\left(2\pi(\frac{2}{32})n\right), \quad n \in Z_{32}$, la cual es una señal sinusoidal, con período fundamental de longitud 16, y sustituir la siguiente gráfica por la gráfica resultante del programa modificado. Note que esta señal discreta o secuencia debe producir dos períodos en la longitud de tiempo asignada. La gráfica resultante debe poseer 32 valores solamente, indexados uniformemente, en incrementos unitarios en el eje de la abscisa y teniendo el "cero" como índice de origen.

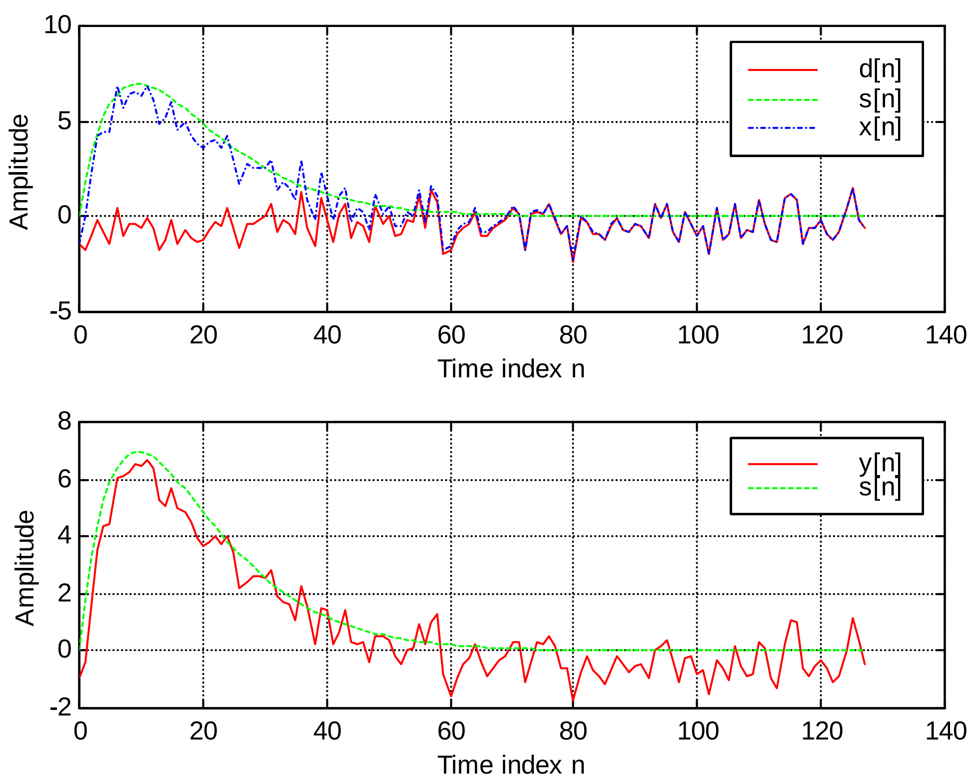
```
% Program P1_4 - Modified by Prof. Domingo Rodriguez - Fall 2003
% Generation of a sinusoidal sequence
n = 0:32;
f = (2/32);
phase = 0;
A = 1.5;
arg = 2*pi*f*n - phase;
x = A*cos(arg);
clf;        % Clear old graph
stem(n,x);      % Plot the generated sequence
axis([0 32 -1.5 1.5]);
grid;
title('Sinusoidal Sequence');
xlabel('Time index n');
ylabel('Amplitude');
axis;
```



Problema No. P1.5: Sistema suavizador de señales

Estudiar el siguiente programa sobre el diseño de un sistema suavizador para tratar a una señal aleatoria afectada por ruido con distribución uniforme. Los sistemas suavizadores son unos de los sistemas más sencillos en diseñar cuando deseamos remover el componente de ruido que afecta a una señal. Modificar el programa para generar ruido con distribución normal, también llamada distribución de Gauss. Sustituir la gráfica que aparece más abajo por la gráfica resultante del programa modificado. Note que la única modificación que se debe hacer al programa dado es en la generación de la señal de ruido. Todo lo demás queda sin afectar.

```
% Program P1_5 - Modified by Prof. Domingo Rodriguez - Fall 2003
% Signal Smoothing by Averaging
clf; R = 128;
d = 0.8*(randn(R,1) - 0.5); % Generate random noise
m = 0:R-1;
s = 2*m.*(0.9.^m); % Generate uncorrupted signal
x = s + d'; % Generate noise corrupted signal
subplot(2,1,1);
plot(m,d','r-',m,s,'g--',m,x,'b-.');
xlabel('Time index n');ylabel('Amplitude');
legend('d[n] ','s[n] ','x[n] '); grid
x1 = [0 0 x]; x2 = [0 x 0]; x3 = [0 x 0 ];
y = (x1 + x2 + x3)/3;
subplot(2,1,2);
plot(m,y(2:R+1),'r-',m,s,'g--');
legend( 'y[n] ','s[n] ');
xlabel('Time index n');ylabel('Amplitude'); grid
```



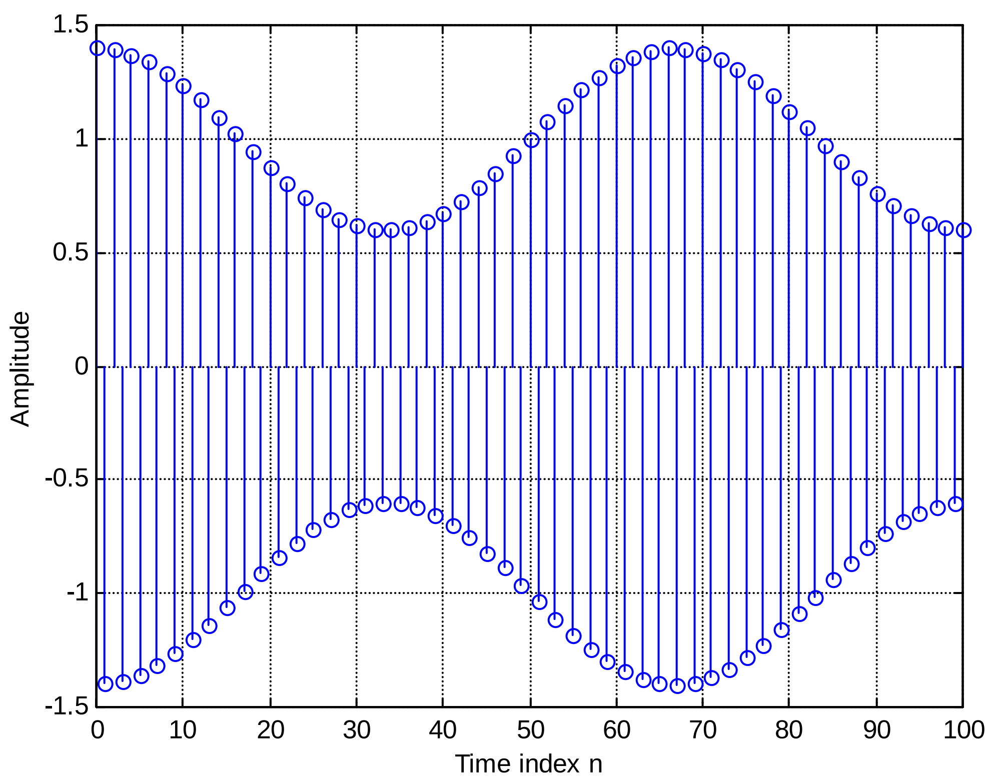
Problema No. P1.6: Modulación lineal

Procesamiento digital de señales es una herramienta muy útil en el modelado y simulación de sistemas de comunicaciones. Por ejemplo, los sistemas de modulación son modelados frecuentemente para estudiar su comportamiento. Este problema simula la modulación lineal de un sistema básico de comunicaciones. Modificar el programa presentado más debajo de la siguiente manera:

1. Sustituir las señales dadas de funciones senos a funciones cosenos
2. Cambiar la frecuencia de la señal portadora fH al valor $fH = 0.5$
3. Cambiar la frecuencia fL a un "**valor único**" en el rango $1 \leq fL \leq 4$.

Sustituir la gráfica que aparece más abajo por la del programa resultante modificado.

```
% Program P1_6 - Modified by Prof. Domingo Rodriguez - Fall 2003
% Generation of amplitude modulated sequence
clf;
n = 0:100;
m = 0.4;fH = 50/100; fL = 1 + (1.5/100);
xH = cos(2*pi*fH*n);
xL = cos(2*pi*fL*n);
y = (1+m*xL).*xH;
stem(n,y);grid;
xlabel('Time index n');ylabel('Amplitude');
```



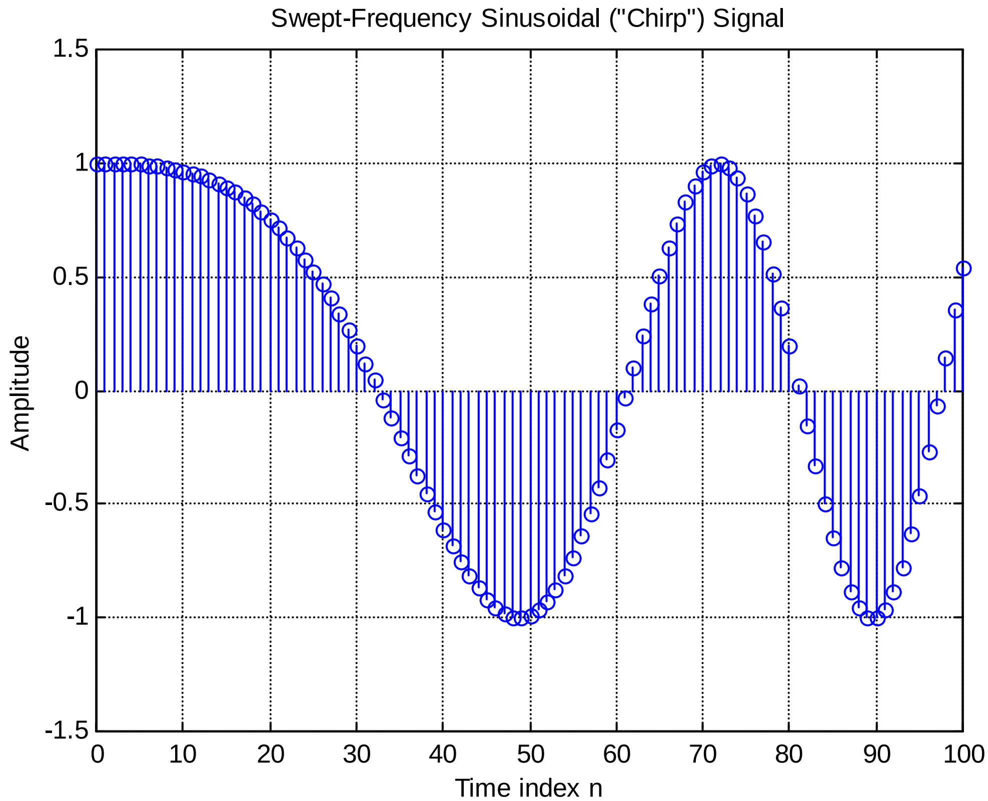
Problema No. P1.7: Señal con barrido lineal de frecuencia

El área de radares con transmisión de señales pulsadas se diseñan pulsos sinusoidales cuya frecuencia cambia con el tiempo, usualmente de forma lineal. Como la frecuencia instantánea es la derivada, con respecto a tiempo, del argumento de una señal sinusoidal, para obtener un barrido lineal en frecuencia es necesario que el argumento de la señal sinusoidal sea una función cuadrática. Modificar el programa presentado más debajo de la siguiente manera:

1. Cambiar el valor del parámetro a a un "**valor único**" en el rango $\frac{1}{1000} \leq a \leq \frac{4}{1000}$
2. Cambiar el valor del parámetro b al valor $b = 2\pi \frac{0.25}{100}$

Sustituir la gráfica que aparece más abajo por la del programa resultante modificado.

```
% Program P1_7 - Modified by Prof. Domingo Rodriguez - Fall 2003
% Generation of a swept frequency sinusoidal sequence
n = 0:100; a = 1/1000;
b = 2*pi*(0.25/100);
arg = a*n.*n + b*n; x = cos(arg);
clf;
stem(n, x);
axis([0,100,-1.5,1.5]);
title('Swept-Frequency Sinusoidal ("Chirp") Signal');
xlabel('Time index n');
ylabel('Amplitude');
grid; axis;
```



Problema No. P1.8: Sistema suavizador de señales

El siguiente programa diseña un sistema suavizador de señales que es una generalización del sistema presentado en el problema No. P1.5 anteriormente. Proceda a correr el programa varias veces sustituyendo la longitud del filtro a través del parámetro M por valores enteros entre en el rango $1 \le M \le 6$. Favor de estudiar los resultados de las corridas y determinar, por inspección visual de las gráficas generadas, cuál de los valores de $1 \le M \le 6$ produce un mejor resultado de filtrado y recuperación de la señal original $s1[n] = \cos(2\pi 0.05n)$, $n \in Z_{101}$. Provea el valor de M seleccionado y una breve narración de la justificación por la selección dentro del espacio enmarcado que aparece más abajo. No es necesario proveer una gráfica del resultado del programa modificado.

Una inspección visual nos inclina a seleccionar al valor de $M = 2$ como el valor que produce menor error entre la señal de salida $y[n]$ y la señal de entrada $s_1[n]$. Una forma más objetiva sería utilizar una métrica para medir el error. Podríamos utilizar, por ejemplo, el valor absoluto de la diferencia entre la señal de salida y la señal de entrada. Esto es, podríamos usar (como una de muchas métricas) la expresión: $|y[n] - s_1[n]|$.

```
% Program P2_1
% Simulation of an M-point Moving Average Filter
% Generate the input signal
n = 0:100;
s1 = cos(2*pi*0.05*n); % A low-frequency sinusoid
s2 = cos(2*pi*0.47*n); % A high frequency sinusoid
x = s1+s2;
% Implementation of the moving average filter
M = input('Desired length of the filter = ');
num = ones(1,M);
y = filter(num,1,x)/M;
% Display the input and output signals
clf;
subplot(2,2,1);
plot(n, s1);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Signal #1');
subplot(2,2,2);
plot(n, s2);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Signal #2');
subplot(2,2,3);
plot(n, x);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Input Signal');
subplot(2,2,4);
plot(n, y);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Output Signal');
axis;
```

Problema No. P1.9: Caracterización de un sistema suavizador de señales como filtro
Utilizar las propiedades que caracterizan a los sistemas lineales e invariantes en el tiempo para demostrar que el sistema suavizador definido por la siguiente ecuación es un filtro:

$$y[n] = T\{x[n]\} = \frac{1}{M}\sum_{k=0}^{M-1} x[n-k]$$

Solución:

Demostración de la condición de linealidad usando homogeneidad y superposición:

Condición de homogeneidad:

$$T\{ax[n]\} = \frac{1}{M}\sum_{k=0}^{M-1} ax[n-k] = a\left(\frac{1}{M}\sum_{k=0}^{M-1} x[n-k]\right) = aT\{x[n]\}$$

Condición de superposición:

$$T\{x_1[n] + x_2[n]\} = \frac{1}{M}\sum_{k=0}^{M-1}\left(x_1[n-k] + x_2[n-k]\right) = \frac{1}{M}\sum_{k=0}^{M-1} x_1[n-k] + \frac{1}{M}\sum_{k=0}^{M-1} x_2[n-k]$$
$$\therefore T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\}$$

Demostración de la condición de sistema invariante: $T\{x[n-n_0]\} = y[n-n_0]$

$$y[n-n_0] = \frac{1}{M}\sum_{k=0}^{M-1} x[\left(n-n_0\right)-k] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-n_0-k]$$

Sea $g[n] = x[n-n_0]$. Procedemos a utilizar la ecuación del sistema

$$T\{x[n-n_0]\} = T\{g[n]\} = \frac{1}{M}\sum_{k=0}^{M-1} g[n-k]$$

$$T\{x[n-n_0]\} = \frac{1}{M}\sum_{k=0}^{M-1} g[n-k] = \frac{1}{M}\sum_{k=0}^{M-1} x[\left(n-k\right)-n_0] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-k-n_0]$$

$$\therefore T\{x[n-n_0]\} = y[n-n_0]$$

Problema No. P1.10: Caracterización de un modulador como sistema lineal

Demostrar que el modulador lineal definido por la ecuación que aparece más abajo no es un filtro.

$$y[n] = T\left\{x[n]\right\} = x[n]\cos(2\pi f_0 n)$$

Solución:

Investigamos si el sistema es invariante a retardos en tiempo. Recordamos que un sistema, sea T, es invariante si satisface la siguiente *condición*:

$$T\left\{x[n-n_0]\right\} = y[n-n_0]$$

i) Utilizamos la ecuación del sistema para describir la parte derecha de la *condición*:

$$y[n-n_0] = x[n-n_0]\cos(2\pi f_0 [n-n_0])$$

Utilizamos la ecuación del sistema para describir la parte izquierda de la *condición*:

Sea $g[n] = x[n-n_0]$, entonces tenemos

$$T\left\{g[n]\right\} = g[n]\cos(2\pi f_0 n)$$

Sustituyendo, terminamos con la siguiente expresión:

$$T\left\{x[n-n_0]\right\} = x[n-n_0]\cos(2\pi f_0 n)$$

$$\therefore T\left\{x[n-n_0]\right\} \neq y[n-n_0]$$