Free extras and updates

1. Updated files

Filename	Reason for update
rms.c	No functional changes. Line-breaks have been adjusted to conform to the listing in the book.
cepstrum.c	No functional changes. Line-breaks have been adjusted to conform to the listing in the book.
cepstral_f0.c	No functional changes. Line-breaks have been adjusted to conform to the listing in the book.
autocorr_f0.c	No functional changes. Line-breaks have been adjusted to conform to the listing in the book.
nfsa1.pl	Error in line 64 (details)

2. An out-take!

The original version of this course used Matlab instead of C for the chapters on signal processing. Although the Matlab sections didn't find its way into the published edition, click here for the original section 3.8, which shows how to make and use IIR filters in Matlab.

3. Extra programs

Filename	Purpose	Usage
2kHz-highpass.c		
2kHz-lowpass.c		
400Hz-lowpass.c		
amplify.c		
dat2txt.c		
depitch.c		
divide.c		

dtw.c		
halve_signal.c		
minus.c		
rms_amplitude.c		
spectral_f0.c		

4. Already-compiled executables (.exe files)

Filename	Purpose	Chapter
autocorr_f0	Estimates f_0 of an input signal using an autocorrelation method.	4
bleep	Compiled from `coswave.c`. Generates a cosine wave file (200 Hz, 1 s "bleep" when played at 8000 samples/s).	2
cepstral_f0	Estimates f_0 of an input signal using a cepstral method.	4
cepstrum	Calculates the cepstrum centred on a specific frame of an input signal.	4
filter	High-pass filters an input signal to produce output file (above 3 kHz at 16,000 samples/s). The filter specification can be altered in the source program, `filter.c`.	3
lpc_spectrum	Calculates a 10-parameter LPC spectrum centred on a specific frame of an input signal (assumed to be 8000 samples/s).	4
lpcana	Analyses an input signal (16,000 samples/s) into a file of 14 linear prediction coefficients every 80 samples and an error signal.	4
lpcsyn	Synthesizes an output signal from a file of linear prediction coefficients and an error signal.	4
meansof4	Filters input signal to produce output file, a running average over a 4-sample window.	3
meansof80		

	Filters input signal to produce output file, a running average over an 80-sample window. (Part solution to exercise 3.3.)	3
multiply	Multiplies two signals, frame-by-frame. (Useful for masking a signal by its voicing estimate, for instance.)	4
normalize	Normalizes the amplitude of an input signal to the range ±32,000 arbitrary units.	4
rms	Calculates overall root mean square amplitude of an input signal.	3
sklatt	Simplified version of the Klatt synthesizer	3
spectrum	Calculates a 512-point FFT power spectral density function centred on a specific frame of an input signal.	4
to_frames	Writes every 80th sample of an input signal to an output file.	4
voicing	Estimates voicing of an amplitude-normalized input signal according to an rms amplitude threshold.	4