**Basic Audio Processing NIM/Nama : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. Membangkitkan sinyal sinus tunggal, menampilkan suara dan menyimpan ke hardisk

Buatlah kode program pada Matlab sebagai berikut.

|  |
| --- |
| clear all;  f=800; %frekuensi sinyal  fs=16000; %frekuensi sampling  t = 0:0.001:1.0; %nilai t mulai dari 0 hingga 1.0 dengan increment 0.001  y=sin(2\*pi\*f\*t);  sound(y,fs) %menampilkan suara  plot(t,y);axis([0 0.1 -1 1]);  title('Sinyal Sinus (f=800 Hz), sampling 16000 Hz') |

1. Jika nilai t diubah menjadi t = 0:1/fs:1.0, apa yang terjadi?
2. Jika f diubah menjadi 262, apa yang terjadi?
3. Jika sinyal y diubah menjadi y = 4\* sin(2\*pi\*f\*t), apa yang terjadi?
4. Jadi, perubahan nilai t akan berpengaruh terhadap \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ sinyal suara; perubahan nilai frekuensi sinyal akan berpengaruh terhadap \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ sinyal suara; dan perubahan ampitudo sinyal akan berpengaruh terhadap \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ sinyal suara.
5. Simpanlah sinyal suara terakhir ke dalam harddisk komputer dengan perintah:

audiowrite(‘sinus1’, y, fs)

Berapa ukuran file yang diperoleh? \_\_\_\_\_\_\_\_\_\_\_\_

1. Menambahkan sinyal offset DC pada sinyal sebelumnya.

|  |
| --- |
| clear all;  f=800; %frekuensi sinyal  fs=16000; %frekuensi sampling  t = 0:0.001:1.0; %nilai t mulai dari 0 hingga 1.0 dengan increment 0.001  y=0.5 +sin(2\*pi\*f\*t);  sound(y,fs) %menampilkan suara  plot(t,y);axis([0 0.1 -1 1]);  title('Sinyal Sinus (f=800 Hz), sampling 16000 Hz') |

1. Apa yang berbeda dengan plot grafik sinyal y=0.5 +sin(2\*pi\*f\*t) ?
2. Apa yang berbeda dengan suara yang dihasilkan dibandingkan sinyal y=0.5 +sin(2\*pi\*f\*t) ?
3. Penggabungan dua buah sinyal suara.

Buatlah file baru, isi dengan kode berikut ini.

|  |
| --- |
| fs = 8000; % frekuensi sampling pada 8KHz  t = 0:1/fs:1; % panjang tiap nada  v1 = 0.5\*cos(2\*pi\*440\*t); % nada A  v2 = 0.5\*cos(2\*pi\*524\*t); % nada lain  v = v1 + v2;  sound(v,fs); % bunyikan pada fs 8kHz  plot(t,v);  title('2 nada')  axis([0 .01 -1 1])  xlabel('Time (sec)')  ylabel('v2(t)') |

Bagaimana suara yang dihasilkan?

1. Buatlah file baru, isi dengan kode berikut ini.

|  |
| --- |
| fs = 8000; % frekuensi sampling pada 8KHz  t = 0:1/fs:2; % panjang tiap nada  f = 500\*t; % frekuensi maksimum 500\*2 = 1000Hz  v = 0.5\*cos(2\*pi\*(f.\*t));  sound(v,fs); % bunyikan pada fs 8kHz  plot(t,v);  title(‘Chirp signal')  axis([0 .01 -1 1])  xlabel('Time (sec)')  ylabel('v(t)') |

1. Bagaimana frekuensi suara yang dihasilkan?
2. Baris keberapa yang menyebabkan suara seperti itu?
3. Bagaimana grafik sinyal yang dihasilkan?
4. Nada dasar dari suara musik didasarkan pada nada A dengan frekuensi 440 Hz. Hitung frekuensi nada-nada yang lain (nada C, D, E, F, G, B, C’) menggunakan rumus berikut:

Dengan:

f = Frekuensi dari nada-nada yang lain

n = langkah (dalam 1 oktaf) dari nada-nada yang akan di cari, dimana urutan nada 1 oktaf adalah ( C, C#, D, D#, E, F, F#, G, G#, A, A#, B)

Tuliskan perhitungannya di bawah ini (tidak hanya hasilnya)

f(C) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f(D) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f(E) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f(F) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f(G) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f(B) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f(C’) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lengkapi kode program berikut dan amati hasilnya.

|  |
| --- |
| fs=8000; % frekuensi sampling 8kHz  t=0:1/fs:0.25; % panjang not ¼ detik  c=sin(2\*pi\*262\*t); % nada c, pembulatan  d=sin(2\*pi\*294\*t); % nada d, pembulatan  e=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; % nada e  f=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; % nada f  g=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; % nada g  a=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; % nada a  b=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; % nada b  c1=\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; % nada c’  nol=zeros(size(t)); % spasi kosong  unyil=[b,a,g,a,b,a,a,nol,b,a,g,e,g,e,e];  sound(unyil,fs); % mainkan pada fs 8kHz  audiowrite('unyil.wav', unyil, fs); % simpan ke file WAV |

1. Jika antara nada di dalam array unyil disisipi dengan nol menjadi unyil=[b,nol,a,nol,g,nol,a,nol,b,...dst], apa yang terjadi?
2. Silakan buat ringtone lain yang sederhana dan singkat ☺. Tuliskan judul ringtone dan arraynya.
3. Merekam suara “aiueo”

Buatlah file baru, isi dengan kode berikut ini.

|  |
| --- |
| clear all;  fs=8000;  suara= audiorecorder(fs, 16, 1); %merekam suara  disp('Start speaking.');  recordblocking(suara,5); %merekam selama 5 detik, ucapkan 'aiueo'  disp('Stop speaking.')  y = getaudiodata(suara);  plot(y);  play(suara);  audiowrite('aiueo.wav',y,fs); |

Perhatikan grafik sinyal yang dihasilkan. Setiap bunyi vokal menampakkan grafik frekuensi tertentu.

1. Memotong sinyal pada bunyi tertentu

Buatlah file baru, isi dengan kode berikut ini.

|  |
| --- |
| clear all;  fs=8000;  y=audioread('aiueo.wav');  suara=audioplayer(y,fs);  figure,plot(y);  %membuat template grafik  A1= min(y);  A2= max(y);  t=length(y);  x1=0; x2=t;  axis([x1 x2 A1 A2]);    %memotong suara ‘O’ yang diperkirakan berada pd t=30200 s/d 33500  y1=y(30200:33500); %sinyal y dicopy pd t=16750 s/d 19500  oo = audioplayer(y1,fs);  play(oo);  figure,plot(y1); |

Range waktu untuk setiap bunyi vokal dilihat berdasarkan grafik sinyal suara file ‘aiueo.wav’.

1. Amati dan catat range waktu untuk bunyi vokal ‘a’, ‘i’, ‘u’, ‘e’, ‘o’

‘a’ pada t = \_\_\_\_\_\_\_\_\_\_\_\_\_ sampai dengan t = \_\_\_\_\_\_\_\_\_\_\_\_

‘i’ pada t = \_\_\_\_\_\_\_\_\_\_\_\_\_ sampai dengan t = \_\_\_\_\_\_\_\_\_\_\_\_

‘u’ pada t = \_\_\_\_\_\_\_\_\_\_\_\_\_ sampai dengan t = \_\_\_\_\_\_\_\_\_\_\_\_

‘e’ pada t = \_\_\_\_\_\_\_\_\_\_\_\_\_ sampai dengan t = \_\_\_\_\_\_\_\_\_\_\_\_

‘o’ pada t = \_\_\_\_\_\_\_\_\_\_\_\_\_ sampai dengan t = \_\_\_\_\_\_\_\_\_\_\_\_

1. Lakukan pemotongan untuk setiap bunyi vokal yang lain selain ‘o’, tambahkan script pada kode di atas untuk pemotongan tersebut.
2. Tuliskan tambahan script tersebut di bawah ini:

|  |
| --- |
| %pemotongan bunyi vokal ‘a’  %pemotongan bunyi vokal ‘i’  %pemotongan bunyi vokal ‘u’  %pemotongan bunyi vokal ‘e’ |