

雑音の生成. genwav.c

```

#include <stdio.h>
#define SIZE (4*44100) /* 練習 1.2 */
int main() {
    FILE *fp2;
    unsigned char
        h[44]={0x52,0x49,0x46,0x46, /* 識別子 RIFF を ascii code で */
              0x84,0x56,0x8,0x0, /* ファイルサイズ-8. [可変] */
              0x57,0x41,0x56,0x45, /* 識別子 WAVE を ascii code で */
              0x66,0x6d,0x74,0x20, /* fmt を ascii code で */
              0x10,0x0,0x0,0x0, /* linear PCM */
              0x1,0x0, /* linear PCM */
              0x2,0x0, /* stereo */
              0x44,0xac,0x0,0x0, /* sampling rate = 44100 = 0xac44 */
              0x10,0xb1,0x2,0x0, /* byte per second, 44100*4 */
              0x4,0x0, /* 16 bit, stereo */
              0x10,0x0, /* bit/sample, 16 bit */
              0x64,0x61,0x74,0x61, /* 識別子 data を ascii code で */
              0xb8,0x55,0x8,0x0}; /* 以下のデータ部分のファイルサイズ. [可変] */

    unsigned char data[SIZE];
    int c;
    int i;
    int filesize = (44+SIZE)-8;
    int datasize = SIZE;
    fp2 = fopen("mysound.wav","w");
    /* SIZE に応じて header 部分のファイルサイズを計算して変更 */
    h[4] = filesize % 0x100; h[5] = (filesize/0x100) % 0x100;
    h[6] = (filesize/0x10000) % 0x100; h[7] = (filesize/0x1000000) % 0x100;
    h[40] = datasize % 0x100; h[41] = (datasize/0x100) % 0x100;
    h[42] = (datasize/0x10000) % 0x100; h[43] = (datasize/0x1000000) % 0x100;

    /* set data in the array data */
    for (i=0; i<SIZE; i++) data[i] = random()%0x100; /* 練習 1.1 */
    /* ファイルの書き出し */
    for (i=0; i<44; i++) fputc(h[i],fp2);
    for (i=0; i<SIZE; i++) fputc(data[i],fp2);
    fclose(fp2);
}
/* WAVE format 参考文献 : http://www.kk.iij4u.or.jp/~kondo/wave/ */

```

練習 1.1. 無音と雑音を繰り返せ.

練習 1.2. 音の長さを変更せよ.

ラの音 (A の音), を左スピーカーから流す. genwav4.c

```
#include <stdio.h>
#include <math.h>
#define SIZE (4*44100*5)
int setL(double hertz,unsigned char data[],int level,int from,int length);

int main() {
    FILE *fp2;
    unsigned char h[44] = {}; /* genwav.c と同じなので略 */
    unsigned char data[SIZE];
    int c;
    int i;
    int filesize = (44+SIZE)-8;
    int datasize = SIZE;
    fp2 = fopen("mysound.wav","w");
    h[4] = filesize % 0x100; h[5] = (filesize/0x100) % 0x100;
    h[6] = (filesize/0x10000) % 0x100; h[7] = (filesize/0x1000000) % 0x100;
    h[40] = datasize % 0x100; h[41] = (datasize/0x100) % 0x100;
    h[42] = (datasize/0x10000) % 0x100; h[43] = (datasize/0x1000000) % 0x100;

    /* set data in the array data */
    for (i=0; i<SIZE; i++) data[i]=0;
    setL(440.0,data,3000,0,4*44100); /* 練習 2.0*/

    for (i=0; i<44; i++) fputc(h[i],fp2);
    for (i=0; i<SIZE; i++) fputc(data[i],fp2);
    fclose(fp2);
}

int setL(double hertz,unsigned char data[],int level,int from,int length) {
    int i,p;
    int w;
    if (4*(from+length) >= SIZE) {
        fprintf(stderr,"Error\n"); return(-1);
    }
    for (i=from; i< from+length; i++) {
        w = (int) level*sin(2*3.14*hertz*((double) i)/44100.0); /*練習 2.2, 2.3*/
        p = i*4;
        /* printf("%d\n",w); */
        if (w < 0) w = w+0x10000;
        data[p] = w % 0x100; /* 練習 2.1 */
        data[p+1] = w/0x100; /* 練習 2.1 */
    }
    return(0);
}
```

練習 2.0. 無音とラの音を繰り返せ.

練習 2.1. 関数 setR を作成し右スピーカーからも同じ音を流せ.

練習 2.2. 和音を生成するには周波数 hertz の異なる sin の和を w とすればよい. ドミソの和音を生成せよ. 音とヘルツの関係は適宜検索.

練習 2.3. 好きな音色をデザイン.