```
// 演習課題5: 圧電スピーカー使用，適当な音階メロディの作成

#include <math.h>  // 算術用ライブラリ

// defineはコンパイル時にマクロ変換される
#define BZ 9   // 圧電スピーカー接続ピン

/**
 * 関数名: setup
 * 引数: なし
 * 処理: 各初期設定のため初回のみ実行
 * 返り値: なし
 */
void setup () {
    // put your setup code here, to run once:

    pinMode ( BZ, OUTPUT );  // 圧電スピーカー接続ピン
    Serial.begin ( 9600 );   // シリアル通信の初期化

    Serial.println("チャルメラ");  // 表示
}

/**
 * 関数名: loop
 * 引数: なし
 * 処理: 無限ループ．任意の周波数で音を交互に鳴らしサイレンを鳴らす
 * 返り値: なし
 */
void loop () {
    // put your main code here, to run repeatedly:
    /**
     * tone関数は3, 11番ピンの出力を妨げる
     */

    // 同メロディ2回
    for (int i = 0; i < 2; i++) {
        tone (BZ, scale2f ( "do" ), 500);
        delay ( 500 );
        tone (BZ, scale2f ( "re" ), 300);
        delay ( 300 );
        tone (BZ, scale2f ( "mi" ), 1000);
        delay ( 1000 );
        tone (BZ, scale2f ( "re" ), 300);
        delay ( 300 );
        tone (BZ, scale2f ( "do" ), 500);
        delay ( 1000 );
    }
    // 最後
    tone (BZ, scale2f ( "do" ), 500);
    delay ( 500 );
    tone (BZ, scale2f ( "re" ), 300);
    delay ( 300 );
    tone (BZ, scale2f ( "mi" ), 500);
    delay ( 500 );
    tone (BZ, scale2f ( "re" ), 1500);
    delay (3000);
}

/**
 * 関数名: scale2f(自作関数)
 * 引数: scale: 音階
 * 処理: 音階から周波数を計算する
 * 返り値: 周波数[Hz]
 */
float scale2f (String scale) {
    float f = 0.0;  // 周波数[Hz]

    // 音階から周波数を決定
    if (scale == "fam1" || scale == "fm1") {
        f = 440.0 * pow(2.0, (-16.0 / 12.0));
    } else if (scale == "fa#m1" || scale == "f#m1") {
        f = 440.0 * pow(2.0, -15.0 / 12.0);
    } else if (scale == "som1" || scale == "gm1") {
        f = 440.0 * pow(2.0, -14.0 / 12.0);
    } else if (scale == "so#m1" || scale == "g#m1") {
        f = 440.0 * pow(2.0, -13.0 / 12.0);
    } else if (scale == "ram1" || scale == "am1") {
        f = 440.0 * pow(2.0, -12.0 / 12.0);
    } else if (scale == "ra#m1" || scale == "a#m1") {
        f = 440.0 * pow(2.0, -11.0 / 12.0);
    } else if (scale == "sim1" || scale == "bm1") {
        f = 440.0 * pow(2.0, -10.0 / 12.0);
    } else if (scale == "do" || scale == "c") {
        f = 440.0 * pow(2.0, -9.0 / 12.0);
    } else if (scale == "do#" || scale == "c#") {
        f = 440.0 * pow(2.0, -8.0 / 12.0);
    } else if (scale == "re" || scale == "d") {
        f = 440.0 * pow(2.0, -7.0 / 12.0);
    } else if (scale == "re#" || scale == "d#") {
        f = 440.0 * pow(2.0, -6.0 / 12.0);
    } else if (scale == "mi" || scale == "e#") {
        f = 440.0 * pow(2.0, -5.0 / 12.0);
    } else if (scale == "fa" || scale == "f") {
        f = 440.0 * pow(2.0, -4.0 / 12.0);
    } else if (scale == "fa#" || scale == "f#") {
        f = 440.0 * pow(2.0, -3.0 / 12.0);
    } else if (scale == "so" || scale == "g") {
        f = 440.0 * pow(2.0, -2.0 / 12.0);
    } else if (scale == "so#" || scale == "g#") {
        f = 440.0 * pow(2.0, -1.0 / 12.0);
    } else if (scale == "ra" || scale == "a") {
        f = 440.0 * pow(2.0, 0.0 / 12.0);
    } else if (scale == "ra#" || scale == "a#") {
        f = 440.0 * pow(2.0, 1.0 / 12.0);
    } else if (scale == "si" || scale == "b") {
        f = 440.0 * pow(2.0, 2.0 / 12.0);
    } else if (scale == "dop1" || scale == "cp1") {
        f = 440.0 * pow(2.0, 3.0 / 12.0);
    } else if (scale == "do#p1" || scale == "c#p1") {
        f = 440.0 * pow(2.0, 4.0 / 12.0);
    } else if (scale == "rep1" || scale == "dp1") {
        f = 440.0 * pow(2.0, 5.0 / 12.0);
    } else if (scale == "re#p1" || scale == "d#p1") {
        f = 440.0 * pow(2.0, 6.0 / 12.0);
    } else if (scale == "mip1" || scale == "ep1") {
        f = 440.0 * pow(2.0, 7.0 / 12.0);
    } else if (scale == "fap1" || scale == "fp1") {
        f = 440.0 * pow(2.0, 8.0 / 12.0);
    } else if (scale == "fa#p1" || scale == "f#p1") {
        f = 440.0 * pow(2.0, 9.0 / 12.0);
    } else if (scale == "sop1" || scale == "gp1") {
        f = 440.0 * pow(2.0, 10.0 / 12.0);
    } else if (scale == "so#p1" || scale == "g#p1") {
        f = 440.0 * pow(2.0, 11.0 / 12.0);
    } else if (scale == "rap1" || scale == "ap1") {
        f = 440.0 * pow(2.0, 12.0 / 12.0);
    } else if (scale == "ra#p1" || scale == "a#p1") {
        f = 440.0 * pow(2.0, 13.0 / 12.0);
    } else if (scale == "sip1" || scale == "bp1") {
        f = 440.0 * pow(2.0, 14.0 / 12.0);
    } else if (scale == "dop2" || scale == "cp2") {
        f = 440.0 * pow(2.0, 15.0 / 12.0);
    }
    return f;  // 周波数を返す
}
```