```
//Halidou Gado Raihanatou
#include <cs50.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>
#include "helpers.h"

int duration(string fraction)
{
    int numerator = fraction[0] - '0';
    int denominator = fraction[2] - '0';
    // This returns the numerator as a single integer
    return ((8/denominator)*numerator);
}

// Calculates frequency (in Hz) of a note

int freq(string note)
{
    int octave = note[strlen(note) - 1] - '0';
    // Getting the frequency for each note type.
    if (note[0] == 'A')
    {
        freq = 440;
    }
    else if (note[0] == 'B')
    {
        freq = 440.0 * (pow(2.0, (2.0 / 12.0)));
    }

    else if (note[0] == 'C')
    {
        freq = 440.0 / (pow(2.0, (9.0 / 12.0)));
    }

    else if (note[0] == 'D')
    {
        freq = 440.0 / (pow(2.0, (7.0 / 12.0)));
    }

    else if (note[0] == 'E')
    {
        freq = 440.0 / (pow(2.0, (5.0 / 12.0)));
```

```
}

else if (note[0] == 'F')
{
    freq = 440.0 / (pow(2.0, (4.0 / 12.0)));
}

else if (note[0] == 'G')
{
    freq = 440.0 / (pow(2.0, (2.0 / 12.0)));
}

// Final adjustment or flat or sharp.
if (note[1] == 'b')
{
    freq /= (pow(2.0, (1.0 / 12.0)));
}
else if (note[1] == '#')
{
    freq *= (pow(2.0, (1.0 / 12.0)));
}

// Loop to shift octave
if (octave > 4)
{
    for (int i = 0; i < octave - 4; i++)
    {
        freq *= 2.0;
    }
}
else if (octave < 4)
{
    for (int i = 0; i < 4 - octave; i++)
    {
        freq /= 2.0;
    }
}

// Final adjustment or flat or sharp
if (note[1] == 'b')
{
    freq /= (pow(2.0, (1.0 / 12.0)));
}
```

```
    else if (note[1] == '#')
    {
        freq *= (pow(2.0, (1.0 / 12.0)));
    }

    // Return frequency value as an int
    int result = round(freq);
    return result;
}


// Determines whether a string represents a rest
bool is_rest(string s)
{
    if (s[0] == '\0')
    {
        return true;
    }

    else
    {
        return false;
    }
}
```