```
    1: // Copyright 2015 <Angel Calcano>
    2: // PS5b
    3: #include <SFML/Graphics.hpp>
    4: #include <SFML/System.hpp>
    5: #include <SFML/Audio.hpp>
    6: #include <SFML/Window.hpp>
    7: #include <math.h>
    8: #include <limits.h>
    9: #include <iostream>
   10: #include <string>
   11: #include <exception>
   12: #include <stdexcept>
   13: #include <vector>
   14: #include "RingBuffer.hpp"
   15: #include "GuitarString.hpp"
   16: #define SAMPLES_PER_SEC 48400
   17:
   18: std::vector< sf::Int16 > makeSamplesFromString(GuitarString *gs) {
   19:   std::vector< sf::Int16 > samples;
   20:   gs->pluck();
   21:   int duration = 8;
   22:   int i;
   23:   for (i= 0; i < SAMPLES_PER_SEC * duration; i++) {
   24:     gs->tic();
   25:     samples.push_back(gs->sample());
   26:   }
   27:   return samples;
   28: }
   29: int main(int argc, char *argv[]) {
   30:   sf::RenderWindow window(sf::VideoMode(300, 200), "SFML Guitar Hero");
   31:   sf::Event event;
   32:   double freq;
   33:   int i;
   34:   std::vector< std::vector< int16_t >  > ado_smpl_strm;
   35:   std::vector< sf::SoundBuffer > ado_smpl;
   36:   std::vector< sf::Sound > SndBffer;
   37:   std::vector< sf::Int16 > samples;
   38:   std::string keyboard = "q2we4r5ty7u8i9op-[=zxdcfvgbnjmk,.;/' ";
   39:   ado_smpl_strm.resize(37);//ALWAYS USE resize
   40:   ado_smpl.resize(37);
   41:   SndBffer.resize(37);
   42:   for (i = 0; i < 37; i++) {
   43:     freq = 220*pow(2, (i-24)/12.0);//changed from 440 to 220
   44:     //GuitarString gs1(freq); making shallow copy
   45:     GuitarString *gs1 = new GuitarString(freq);
   46:     samples = makeSamplesFromString(gs1);
   47:     ado_smpl[i].loadFromSamples( &samples[0], samples.size(), 2, SAMPLES_PER
_SEC); // y
   48:     SndBffer[i].setBuffer(ado_smpl[i]);
   49:     //sf::SoundBuffer buf1;
   50:     //sf::Sound sound1;
   51:     //if (!buf1.loadFromSamples(&ado_smpl_strm[i][0], ado_smpl_strm[i].size(
), 2, SAMPLES_PER_SEC))
   52:     //throw std::runtime_error("sf::SoundBuffer: failed to load from samples
.");
   53:     //sound1.setBuffer(buf1);
   54:     //ado_smpl.push_back(buf1);
   55:     //sound1.setBuffer(ado_smpl[i]);
   56:     //SndBffer.push_back(sound1);
   57:   }
   58:   int index;
```

```
   59:   while (window.isOpen()) {
   60:     while (window.pollEvent(event)) {
   61:       switch (event.type) {
   62:       case sf::Event::Closed:
   63:         window.close();
   64:         break;
   65:       case sf::Event::TextEntered:
   66:         index = keyboard.find(event.text.unicode);
   67:         if ((unsigned)index != std::string::npos){
   68:            SndBffer[index].play();
   69:         }
   70:         break;
   71:       default:
   72:         break;
   73:       }
   74:     }
   75:     window.clear();
   76:     window.display();
   77:   }
   78:   return 0;
   79: }
```