Lab 6 - Metronome

Alexander Stradnic - 119377263

I set up the LCD, buttons (from left to right in pictures, Tuning/Metronome toggle, slower, and faster), and buzzer. I used the built in green LED to flash when the metronome is on, and be constant when in Tuning mode.

Code

```cpp
#include "mbed.h"
#include "platform/mbed_thread.h"

#include "TextLCD.h"

TextLCD lcd(D12, D11, D5, D4, D3, D2);

#define ConcertA 440
#define DefaultTempo 88
#define Minute 60000
#define HalfMinute 30000

DigitalOut led(LED1);
DigitalOut buzzer(D9);

InterruptIn lessButton(D6);
InterruptIn moreButton(D7);
InterruptIn tuneToggle(D8);

int tempo = DefaultTempo;
bool tuning = false;
bool tPrint = false;
int note;

void playNote(int f){ // Play note of frequency f, using while true loop
for continuous sound
    float t = (1000/f)/2;
    thread_sleep_for(t);
    buzzer = !buzzer;
    thread_sleep_for(t);
}
```

```cpp
void pTempo(){
    lcd.cls();
    lcd.printf("Tempo:%dbpm", tempo);
}

void reduce(){
    tempo -= 4;
}

void increase(){
    tempo += 4;
}

void setTuning(){ // Toggle tuning and tPrint. tPrint is used so that the
LCD doesn't have to refresh through every loop iteration
    tuning = !tuning;
    tPrint = tuning ? true : false;
}

int main()
{

    lessButton.rise(&reduce);
    moreButton.rise(&increase);
    tuneToggle.rise(&setTuning);

    while (true) {
    if(tuning){
        if(tPrint){
            lcd.cls();
            lcd.printf("Tuning:%dHz", ConcertA);
            tPrint = false;
            led = 1;
            note = ConcertA;
        }
        playNote(note);
    }
    else{
        pTempo();

        int t = (HalfMinute/tempo) - 10;
```

```
            thread_sleep_for(t);
            led = !led;
            for(int i = 100; i > 0; i--){ // This loop acts produces a
short "tick" sound from the buzzer
                buzzer = !buzzer;
                thread_sleep_for(0.1f);
            }

            thread_sleep_for(t);
            led = !led;
        }
        }
}
```

Pictures