```cpp
// pixels.cpp:
// using SFML to load a file, manipulate its pixels, write it to disk
// Fred Martin, fredm@cs.uml.edu, Sun Mar  2 15:57:08 2014

// g++ -o pixels pixels.cpp -lsfml-graphics -lsfml-window

#include <SFML/System.hpp>
#include <SFML/Window.hpp>
#include <SFML/Graphics.hpp>
#include "LFSR.hpp"


int main(int argc, char* argv[])
{
        string input_file = argv[1];
        string output_file = argv[2];
        string seed = argv[3];
        int tap = atoi(argv[4]);

        sf::Image first;
        if (!first.loadFromFile(input_file))
                return -1;

        sf::Image second;
        if (!second.loadFromFile(input_file))
                return -1;

        // p is a pixel
        sf::Color p;
        sf::Vector2u win1_size = first.getSize();

        LFSR lfsr(seed, tap);

        // create encrypted image of the original image
        for (unsigned int x = 0; x < win1_size.x; x++) {
                for (unsigned int y = 0; y < win1_size.y; y++) {
                        p = second.getPixel(x, y);
                        p.r = p.r ^ lfsr.generate(5);
                        p.g = p.g ^ lfsr.generate(5);
                        p.b = p.b ^ lfsr.generate(5);
                        second.setPixel(x, y, p);
                }
        }

        sf::RenderWindow window1(sf::VideoMode(win1_size.x, win1_size.y), "First");
        sf::RenderWindow window2(sf::VideoMode(win1_size.x, win1_size.y), "Second");

        sf::Texture original;
        original.loadFromImage(first);
        sf::Texture encrypted;
        encrypted.loadFromImage(second);

        sf::Sprite sprite1;
        sprite1.setTexture(original);
        sf::Sprite sprite2;
        sprite2.setTexture(encrypted);

        while (window1.isOpen() && window2.isOpen()) {
        sf::Event event;
```

```
                while (window1.pollEvent(event)) {
                        if (event.type == sf::Event::Closed)
                                window1.close();
                        else if (sf::Keyboard::isKeyPressed(sf::Keyboard::Escape))
                                window1.close();
                }
                while (window2.pollEvent(event)) {
                        if (event.type == sf::Event::Closed)
                                window2.close();
                        else if (sf::Keyboard::isKeyPressed(sf::Keyboard::Escape))
                                window2.close();
                        }
                window1.clear();
                window1.draw(sprite1);
                window1.display();
                window2.clear();
                window2.draw(sprite2);
                window2.display();
        }

        // fredm: saving a PNG segfaults for me, though it does properly
        //   write the file
        if (!second.saveToFile(output_file))
                return -1;

        return 0;
}
```