proj1a.cpp

```
#include "opencv2/highgui.hpp"
#include <iostream>
using namespace cv;
using namespace std;

int main(int argc, char** argv) {
  if(argc != 3) {
    cout << argv[0] << ": "
         << "got " << argc-1 << " arguments. Expecting two: width height."
         << endl ;
    return(-1);
  }

  int width = atoi(argv[1]);
  int height = atoi(argv[2]);
  int** RED1 = new int*[height];
  int** GREEN1 = new int*[height];
  int** BLUE1 = new int*[height];
  int** RED2 = new int*[height];
  int** GREEN2 = new int*[height];
  int** BLUE2 = new int*[height];

  for(int i = 0 ; i < height ; i++) {
    RED1[i] = new int[width];
    GREEN1[i] = new int[width];
    BLUE1[i] = new int[width];
    RED2[i] = new int[width];
    GREEN2[i] = new int[width];
    BLUE2[i] = new int[width];
  }

  for(int i = 0 ; i < height ; i++)
    for(int j = 0 ; j < width ; j++)
      {
        int r1, g1, b1;
        int r2, g2, b2;

        double x = (double)j/(double)width;
        double y = (double)i/(double)height;
        double Y = 1.0;

        double L = 90;
        double u = x * 512 - 255;
        double v = y * 512 - 255;


        /* Your code should be placed here
           It should translate xyY to byte sRGB
           and Luv to byte sRGB
        */
        r1 = (int) (x * 255);
        g1 = (int) (y * 255);
        b1 = (int) (1.0 * 255);

        r2 = (int) (1.0 * 255);
        g2 = (int) (x * 255);
        b2 = (int) (y * 255);

        // this is the end of your code

        RED1[i][j] = r1;
        GREEN1[i][j] = g1;
        BLUE1[i][j] = b1;
        RED2[i][j] = r2;
        GREEN2[i][j] = g2;
        BLUE2[i][j] = b2;
```

```
67        }
68
69
70    Mat R1(height, width, CV_8UC1);
71    Mat G1(height, width, CV_8UC1);
72    Mat B1(height, width, CV_8UC1);
73
74    Mat R2(height, width, CV_8UC1);
75    Mat G2(height, width, CV_8UC1);
76    Mat B2(height, width, CV_8UC1);
77
78    for(int i = 0 ; i < height ; i++)
79      for(int j = 0 ; j < width ; j++) {
80
81        R1.at<uchar>(i,j) = RED1[i][j];
82        G1.at<uchar>(i,j) = GREEN1[i][j];
83        B1.at<uchar>(i,j) = BLUE1[i][j];
84
85        R2.at<uchar>(i,j) = RED2[i][j];
86        G2.at<uchar>(i,j) = GREEN2[i][j];
87        B2.at<uchar>(i,j) = BLUE2[i][j];
88      }
89
90    Mat xyY;
91    Mat xyY_planes[] = {B1, G1, R1};
92    merge(xyY_planes, 3, xyY);
93    namedWindow("xyY",CV_WINDOW_AUTOSIZE);
94    imshow("xyY", xyY);
95
96    Mat Luv;
97    Mat Luv_planes[] = {B2, G2, R2};
98    merge(Luv_planes, 3, Luv);
99    namedWindow("Luv",CV_WINDOW_AUTOSIZE);
100   imshow("Luv", Luv);
101    waitKey(0); // Wait for a keystroke
102   return(0);
103 }
```