CS5820: Graphical Processing Unit
SW: Assignment 3

Abburi Venkata Sai Mahesh - CS18BTECH11001

December 27, 2020

This document is generated by LaTeX

Problem Statement:
Develop an optimized algorithm to render a triangle without using any graphics library.

Given Specifications:
Image width = 400
Image height = 200

Background Colour = Black[0,0,0]
Triangle Colour = Red[1,0,0]

Coordinates of the triangle: $\{\{200, 180\}, \{100, 40\}, \{300, 40\}\}$
Interpolation Equations of the triangles

Line 1: $\{\{200, 180\}, \{100, 40\}\}$

$$\begin{aligned} \frac{y-180}{x-200} &= \frac{40-180}{100-200} \\ \frac{y-180}{x-200} &= 1.4 \\ y-180 &= 1.4x-280 \\ &\boxed{\therefore y = -100+1.4x} \end{aligned}$$

Line 2: $\{\{100, 40\}, \{300, 40\}\}$

$$\begin{aligned} \frac{y-40}{x-100} &= \frac{40-40}{300-100} \\ \frac{y-40}{x-100} &= 0 \\ y-40 &= 0 \\ &\boxed{\therefore y = 40} \end{aligned}$$

Line 3: $\{\{300, 40\}, \{200, 180\}\}$

$$\begin{aligned} \frac{y-40}{x-300} &= \frac{180-40}{200-300} \\ \frac{y-40}{x-300} &= -1.4 \\ y-40 &= -1.4x+420 \\ &\boxed{\therefore y = 460-1.4x} \end{aligned}$$

Program(`CS5820_SW3.cpp`):

```cpp
#include <bits/stdc++.h>
using namespace std;

int main () {
  // Setting constants for the image details
  const int image_width = 400;
  const int image_height = 200;

  cout << "P3\n" << image_width << "␣" << image_height << "\n255\n";

  // Creating the triangle image
  // Starting from the bottom of the image
  for (int j = image_height-1; j >= 0; --j) {
    // Starting from left to right
    for (int i = 0; i < image_width; ++i) {

      // Setting initial colour to black
      auto r = 0;
      auto g = 0;
      auto b = 0;

      // Lablelling the portion inside traingle to red
      // Using the interpolation equations of the lines
      // of edges of the triangle
      if (j > 40 && j < -100 + i*1.4 && j < 460 + i*(-1.4))
        r = 1;

      // Loading the rgb values of the figure
      int ir = static_cast<int>(255.999*r);
      int ig = static_cast<int>(255.999*g);
      int ib = static_cast<int>(255.999*b);

      // Printing onto the screen
      cout << ir << "␣" << ig << "␣" << ib << endl;
    }
  }
}
```

Compiling:
The above program can be compiled Using the below command

```
$ g++ -o SW3 CS5820_SW3.cpp
```

Execution:
To run the above program run the below command

```
$ ./SW3 >triangle.ppm
```

This will generate the image in the same directory with the name `triangle.ppm`

`triangle.ppm`
The final image of triangle genrated by above code is presented here:

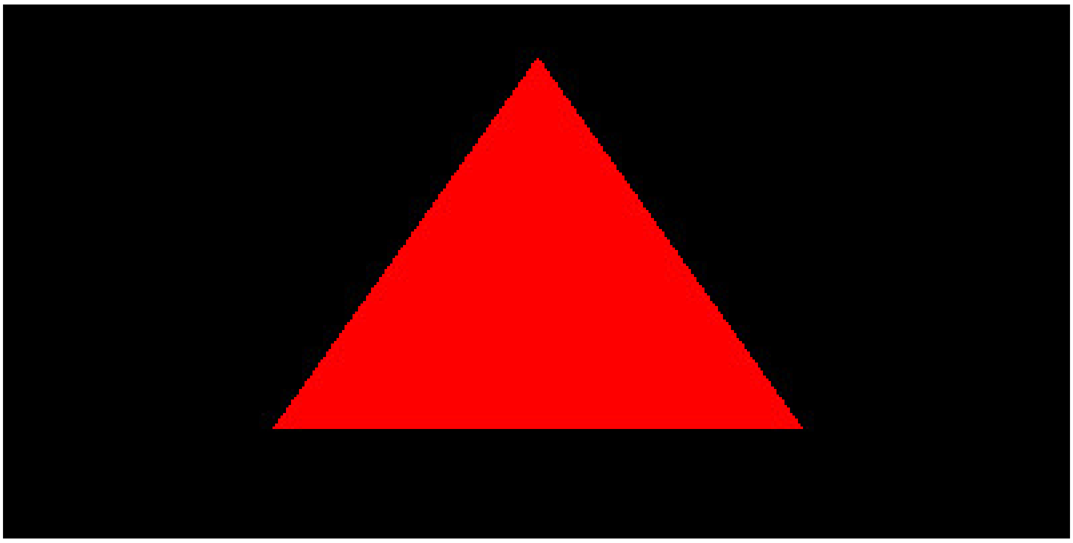

Figure 1: triangle.ppm