

# **Informe Escrito**

Parcial 2 - Primera parte

**Julian Taborda Ramirez**

**Samuel Ruiz Vargas**

Informatica II  
Universidad de Antioquia  
Medellín  
Septiembre de 2021

# Índice

<b>1. Análisis del problema</b>	<b>2</b>
1.1. Objetivo(s) . . . . .	2
1.2. Herramientas . . . . .	2
1.3. Problemática(s) . . . . .	2
<b>2. Esquema de desarrollo algoritmico</b>	<b>2</b>
2.1. QT . . . . .	2
2.2. Tinkercad . . . . .	3
<b>3. Algoritmo implementado</b>	<b>3</b>
3.1. QT . . . . .	3
3.2. Tinkercad . . . . .	3
<b>4. Consideraciones</b>	<b>3</b>

## **1. Análisis del problema**

### **1.1. Objetivo(s)**

Nuestro objetivo es simple, necesitamos ser capaces de representar una imagen en una matriz de leds de un tamaño por determinar, de la forma más fiel posible e independiente de la resolución de la imagen, haciendo uso de las técnicas de submuestreo y sobremuestreo.

### **1.2. Herramientas**

Como herramientas de hardware tenemos las tiras neopixel, la tarjeta Arduino uno R3. Además probamos las funciones que nos proporcionaron para analizar una imagen.

### **1.3. Problemática(s)**

Una de las problemáticas a la que nos enfrentamos es la forma en la que vamos a tratar la información extraída de la imagen en QT y posteriormente utilizar esa información en tinkercad.

Otra de las problemáticas que tenemos es como redimensionar la imagen, de tal manera que, se adapte a la matriz independientemente a su relación de aspecto.

## **2. Esquema de desarrollo algorítmico**

### **2.1. QT**

El código de QT debe tener la capacidad de recibir una imagen y ser capaz de entregarnos la información necesaria para el desarrollo de la totalidad de nuestro algoritmo; Esta es nuestra prioridad en la etapa mas temprana de nuestro desarrollo y sobre la que se debe basar nuestro proceso.

Una vez conseguida la primera parte del algoritmo tenemos que proceder a analizar de que forma vamos a tratar la información sustraída de dicha imagen, con el fin de facilitar la lectura una vez en Tinkercad, dicho esto el manejo de tipos de datos será esencial para el correcto desempeño del programa.

## **2.2. Tinkercad**

Para el desarrollo del código en Tinkercad, pensamos en el flujo de la información de tal manera que podamos recibir la información de QT (La cual esta previamente procesada para no causar conflictos con Tinkercad) y procesarla según dicte el algoritmo que implementemos para realizar la representación de la matriz de Neopixel.

## **3. Algoritmo implementado**

### **3.1. QT**

Para el algoritmo implementado en QT, pensamos en hacer uso de las librerías estandar de QT, para leer y escribir una imagen de tal manera que la información obtenida pueda ser manipulada. De esta manera primero validamos la existencia de la imagen, pidiendo la dirección de esta, luego se procede a cargar con la función `ImageRead()` y escribimos sus datos con `ImageWrite()`, finalmente guardamos la información en un archivo de texto que posteriormente será utilizado para tratar la información en Tinkercad.

### **3.2. Tinkercad**

En Tinkercad tomaremos la información correspondiente de la imagen guardada en un archivo de texto, aplicando alguna técnica de muestreo (Submuestreo o Sobremuestreo) podremos escalar la imagen de tal manera que se adecue a la cantidad de leds que tenemos en la matriz. Claramente debemos tener en cuenta el escalamiento correspondiente ya que debemos mantener una fidelidad con la relación aspecto de la imagen.

## **4. Consideraciones**

Debemos de tener en cuenta la cantidad de leds que tenemos en la matriz ya que, dependiendo de la cantidad de píxeles de la imagen se tiene que aplicar cualquiera de las dos técnicas, para adaptar la imagen a la cantidad de leds de la matriz.