

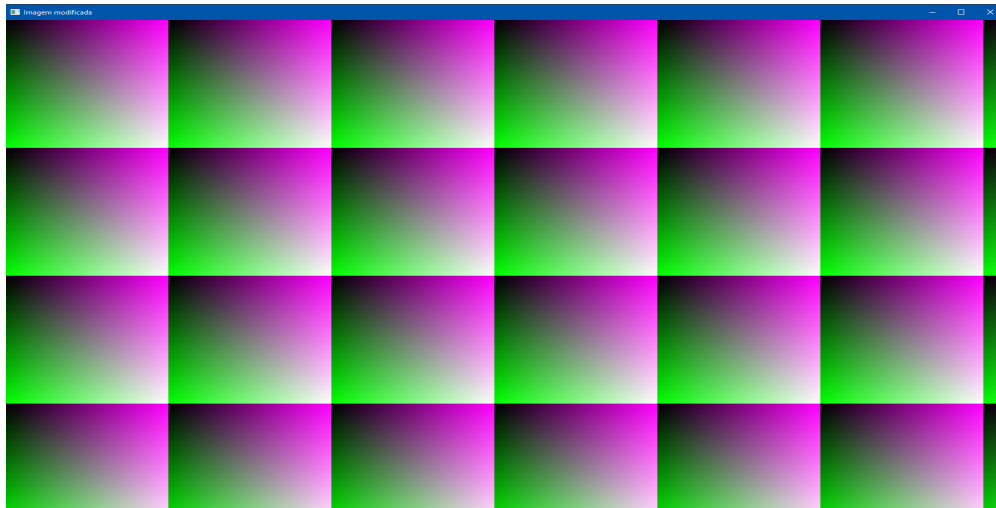


Universidade Federal de Campina Grande
João Vitor de Melo Cavalcante e Souza

Para a realização da Etapa 2, acompanhei de perto a apostila fornecida. Pela minha extensa familiaridade com Python, pude desfrutar de perto os exemplos e explicações da mesma.

Utilizei o pacote mais recente fornecido pela extensão pip e fui para o aprendizado. A apostila direciona muito bem por meio de exemplos que mostram as funcionalidades básicas e o modus operandi da biblioteca. Dividi o conjunto de códigos dados em respectivos arquivos .py. **No primeiro** (exemplo1.py) a apostila demonstra como a biblioteca pode receber uma imagem e demonstrar dados acerca dela como: Altura, Largura e Quantidade de Canais. O programa então, além de demonstrar os dados mencionados anteriormente, salva uma cópia da imagem fornecida na entrada como “saida.jpg”.

No segundo exemplo, respectivo ao capítulo 2 (exemplo2.1.py), foi feito um programa bastante simples que seria ampliado e aprofundado nos subsequentes exemplos. Tal programa checa, a “quantidade” de cada uma das três cores do sistema RGB (**vermelho**, **verde** e **azul**) do pixel [0,0], isto é, no primeiro pixel da esquerda para a direita, de cima para baixo, utilizando o sistema de matriz para cada um dos pixels da foto. Como dito anteriormente, o exemplo acabaria por ser aprofundado em um **terceiro exemplo** (intitulado como exemplo2.2.py). Neste exemplo é demonstrado a capacidade da biblioteca de alterar uma imagem e demonstrar tal alteração, modificando todos os pixels na imagem para uma cor totalmente azul (255,0,0 na biblioteca), o que foi subsequentemente alterado para um exemplo (exemplo2.3.py) em que os índices do laço que percorria os pixels da imagem influenciaria a “cor” a ser atribuída para os mesmos pixels.



De forma similar, o **quarto** exemplo (exemplo2.4.py) utiliza os índices mas apenas para uma cor, sendo esta o verde, em um cálculo ($x*y\%256$) que gerava uma “bela imagem”, assim chamada pela apostila. Tanto no exemplo 3 como no 4, as alterações nas cores são chamadas de “alterações dinâmicas”. No **quinto** e ultimo exemplo (exemplo2.5.py), a apostila demonstra como, utilizando além das chamadas alterações dinâmicas, técnicas referentes ao iterador em um laço pode ajudar a formar desenhos, ou como no caso discutido, pontilhados por toda a imagem. “Pulando” de 10 em 10 unidades no percorrimento da imagem, nas suas coordenadas, o programa altera a cor de cada pixel durante tais saltos para amarelo, resultando em uma imagem “pontilhada”, com quadrados 10x10.

