

Vítor Paixão Damasceno

Prática 15

Laboratório de AEDS

Belo Horizonte, Brasil

2024

1 Introdução

Neste laboratório, exploramos a representação de cores em uma grade infinita utilizando um dicionário. O código fornecido inicialmente mostrava uma grade infinita com todas as células brancas. O usuário pode movimentar a grade arrastando o mouse e dar zoom com a roda do mouse. A grade é representada por um dicionário, onde a chave é uma string no formato "(linha,coluna)" e o valor é a cor. No entanto, a cor branca não é armazenada.

O objetivo desta prática foi modificar o código para que, ao clicar em uma célula, ela altere sua cor de acordo com o número de cliques (preto, verde, vermelho, azul, amarelo e branco), de maneira circular.

2 Desenvolvimento

Para adicionar mais estados às células da grade, foi necessário alterar a função 'mouseReleased()', especificamente dentro da condição ativada ao apertar o botão esquerdo do mouse.

```
1 void mouseReleased() {
2     if (!estaArrastando) {
3         float mundoX = (mouseX - width/2 + deslocamentoX) / zoom;
4         float mundoY = (mouseY - height/2 + deslocamentoY) / zoom;
5
6         int gradeX = floor(mundoX / tamanhoCelula);
7         int gradeY = floor(mundoY / tamanhoCelula);
8
9         String chave = gradeX + "," + gradeY;
10
11        if (mouseButton == LEFT) {
12            if (coresCelulas.containsKey(chave)){
13                int antigaCor = coresCelulas.get(chave);
14                int novaCor = color(0);
15
16                switch(antigaCor) {
17                    case color(255):
18                        novaCor = color(0);
19                        break;
20                    case color(0, 255, 0):
21                        novaCor = color(255, 0, 0);
22                        break;
23                    case color(0):
24                        novaCor = color(0, 255, 0);
```

```
25         break;
26     case color(255, 0, 0):
27         novaCor = color(0, 0, 255);
28         break;
29     case color(0, 0, 255):
30         novaCor = color(255, 255, 0);
31         break;
32     case color(255, 255, 0):
33         coresCelulas.remove(chave);
34         break;
35     }
36
37     coresCelulas.put(chave, novaCor);
38 } else {
39     coresCelulas.put(chave, color(0));
40 }
41 }
42
43 redraw();
44 }
45 }
```

Listing 1 – Função mouseReleased

A variável 'antigaCor' foi criada para armazenar o estado atual da célula e, com base nela, atribuir a nova cor. Além disso, se a cor for amarela, a célula é removida da grade.

3 Resultados

Após as alterações no código, obtivemos os seguintes resultados:

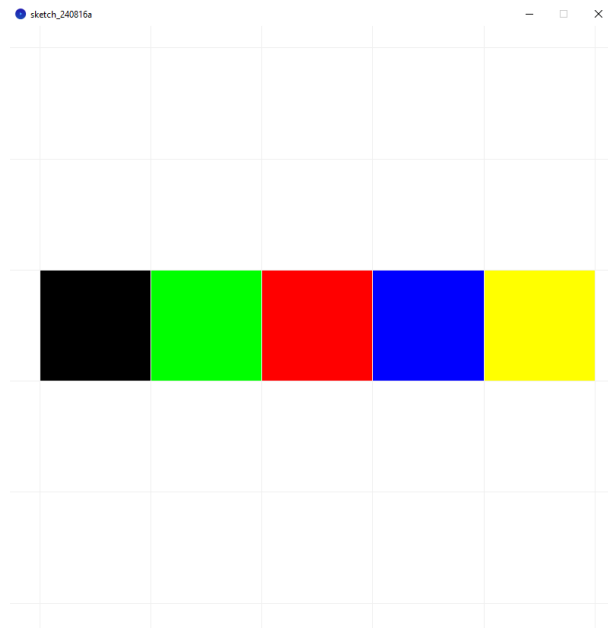


Figura 1 – As 5 possíveis cores de um pixel.

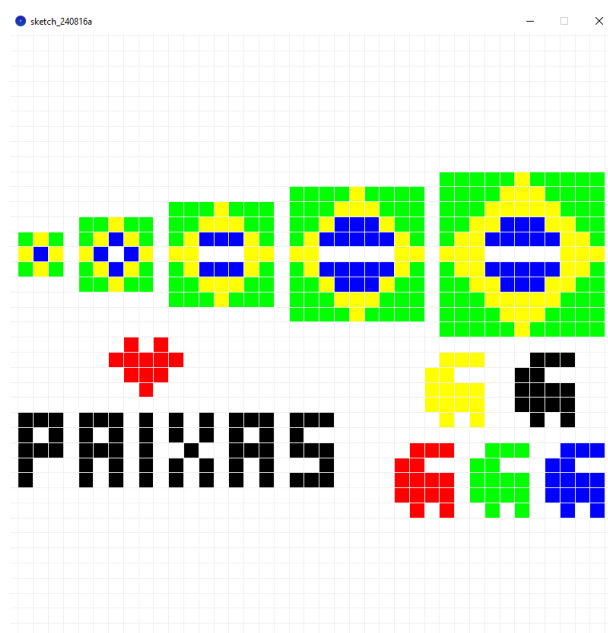


Figura 2 – Desenhos criados com as células coloridas.

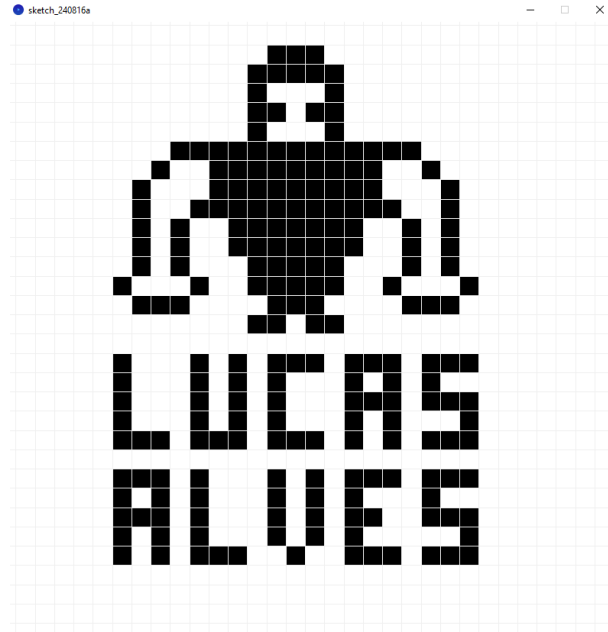


Figura 3 – Caricatura feita com as células coloridas.

4 Conclusão

Os objetivos do experimento foram alcançados com sucesso. A análise do cuidadosa do código permitiu um entendimento mais profundo sobre o uso de dicionários, possibilitando concluir o desafio sem grandes dificuldades.