



Universidade Federal de Ouro Preto - UFOP  
Disciplina: BCC 326 - Processamento de Imagens  
Professor: Guillermo Cámara-Chávez

1.85  
b

Aluno: Cauce Silas No. 21.1.1111

A cola não será tolerada. Se alguém for pego colando, será reprovado com Zero. É considerado cola: olhar/copiar da prova de outro ou deixar outro aluno olhar sua prova.

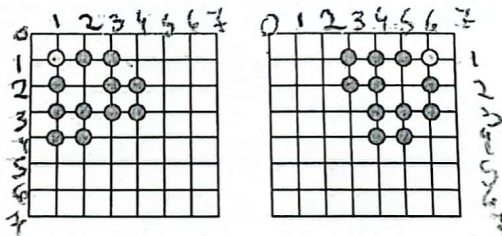
## 2da. Avaliação

1. (2 pts) Na primeira avaliação, foi implementado o tratamento de bordas em uma imagem por meio da operação de *padding*, utilizando as técnicas de *zero-padding* e replicação de elementos. Agora, vamos estender esse processo, implementando as técnicas de *padding simétrico* e *padding simétrico circular*.

Implemente uma função que realize esses dois tipos de padding para imagens, permitindo a aplicação de filtros de qualquer tamanho.

2. (1 pt) Enumere e descreva quais parâmetros o usuário deve definir para executar uma Rede Neural MLP.

3. (1 pt) Encontrar o código de cadeia e a diferença. Use uma vizinhança 4



4. (2 pts) Efetue a seguinte operação de convolução sobre volume, considerando *padding* igual a zero e *stride* igual a um. À esquerda, estão os valores dos três canais da entrada, e à direita, os valores dos três canais do filtro. Escrever todas as operações na prova.

1	2	0
1	1	1
2	2	3

2	1
1	2

0	1	0
1	3	1
2	4	3

1	2
2	1

2	0	0
1	1	1
2	2	3

2	0
0	2

5. (2 pts) Responda o que é executado em cada uma das linhas desde a linha 6 à linha 13 do seguinte código:

```
1 def fit(model, criterion, optimizer, train_loader, test_loader, epochs):
2     train_losses = np.zeros( epochs )
3
4     for it in range(epochs):
5         train_loss = []
6         for inputs, targets in train_loader:
7             inputs, targets = inputs.to(device), targets.to(device)
8             optimizer.zero_grad()
9             outputs = model(inputs)
10            loss = criterion(outputs, targets)
11            loss.backward()
12            optimizer.step()
13            train_loss.append( loss.item() )
14
15            train_loss = np.mean(train_loss)
16            train_losses[it] = train_loss
17    return train_losses
```

6. (1 pt) Explique o funcionamento de um **Dataloader** e quais parâmetros devem ser fornecidos
7. (1 pt) Explique a diferença entre gradiente descente batch, gradiente descente estocástico e gradiente descente mini-batch