

	<b>Universidade Federal do ABC</b> <b>Bacharelado em Ciência da Computação</b> <b>Disciplina:</b> Processamento Digital de Imagens <b>Prof.:</b> Francisco Zampiroli <b>Turma:</b> Imagem 2020 <b>Exame:</b> Atividade 4	<b>Sala:</b> 123 <b>Data:</b> 17-05-2020
	<b>Ass.:</b> _____ <b>Estudante:</b> Marcelo Pena <b>ID/RA:</b> 11039314	



#138 - 2020-05-01 - 13:02:22

**Instruções:**

- Esta é a Atividade 4 do ECE, para ser enviada pelo Moodle.
- Esta é uma atividade individual, foram geradas mais de 50 variações desta questão e cada aluno vai receber uma questão distinta.
- Sugestão: resolver o problema primeiro para dimensões pequenas, para facilitar a validação do seu código.
- Antes de submeter, valide o seu código em IDE's com Jupyter Notebook, ou <https://repl.it/languages/python3>

**Questões Dissertativas:**

- Considere a dilatação definida por:

$$dil(f, b) = f \oplus b = \delta_b(f) = \max\{f(y) + b(y - x) : y \in \mathbb{B}_x \cap \mathbb{E}, \forall x \in \mathbb{E}\}$$

Considere também a erosão definida por:

$$ero(f, b) = f \ominus b = \varepsilon_b(f) = \min\{f(y) - b(x - y) : y \in \mathbb{B}_x \cap \mathbb{E}, \forall x \in \mathbb{E}\}$$

Onde  $f \in K^{\mathbb{E}}$  ou  $f \in [0, k]^{\mathbb{E}}$ ,  $k$  é um inteiro positivo representando os níveis de cinza da imagem digital com domínio  $\mathbb{E}$ ,  $b \in \mathbb{Z}^{\mathbb{B}}$ . Considere também  $b \in \mathbb{Z}^{\mathbb{B}}$  a função estruturantes (vizinhança/kernel), conforme exemplo abaixo, com origem sendo o seu centro. Considere o deslocamento de vizinhança igual a atividade VPL aula02-ex1 - erosion.

Implemente a sua solução para o **operador morfológico** a seguir:

$$ero(f, b) = f \ominus b = \varepsilon_b(f)$$

Exemplos de operadores morfológicos: [https://docs.opencv.org/trunk/d9/d61/tutorial\\_py\\_morphological\\_ops.html](https://docs.opencv.org/trunk/d9/d61/tutorial_py_morphological_ops.html).

**ATENÇÃO:** Submeter o arquivo **Q1.py** (com a resposta).

**Exemplo (considerar somente os números como elementos de entrada/saída para os casos de teste):**

```

b:
0 0 1 1
1 0 0 0
0 0 1 0
f:
0 4 6 0 4 4
0 4 2 6 2 4
2 6 6 2 0 2
2 0 2 2 6 0
6 6 2 6 0 0
4 4 0 2 0 0
8 2 2 4 4 4
resultado:
0 -1 0 0 -1 2
0 -1 -1 -1 0 0
-1 0 0 0 -1 -1
0 0 -1 -1 -1 0
-1 -1 0 -1 -1 -1
0 0 -1 -1 -1 -1
-1 -1 -1 -1 -1 0

```