

# Roteiro de Experiência

## 1. Desenvolvimento de Plugin

A programação de rotinas simples em Java.  
Abra o pacote com plugins e imagens pointwise.zip.  
Acesso fácil a pixéis de imagem: classe ImageAccess.

Nome	Descrição	Exemplo
ImageAccess(nx, ny)	Construtor: cria uma imagem de tamanho [nx,ny]	<code>ImageAccess im = new ImageAccess (nx, ny);</code>
getPixel(x, y)	Retorna o valor do pixel na posição (x, y)	<code>value = im.getPixel(x, y);</code>
putPixel(x, y, value)	Atribui value para o pixel (x, y)	<code>im.putPixel(x, y, value);</code>
getWidth() getHeight()	Retorna a largura de uma imagem Retorna a altura de uma imagem	<code>int nx = im.getWidth(); int ny = im.getHeight();</code>
getMaximum() getMinimum()	Retorna o máximo de uma imagem Retorna o mínimo de uma imagem	<code>double max = im.getMaximum(); double min = im.getMinimum();</code>

## 2. A transformação pontual para melhorar visualização.

### 2.1. Entendimento: inversão de contraste.

Entenda a rotina `inverse()`, ela muda os valores de pixel da imagem  $f(x, y)$  para  $g(x, y) = 255 - f(x, y)$ .

Aplique a rotina à imagem `microtubules.tif` chamando o plugin "Inverse".  
Observe o histograma.

2.2. Esticando e normalizando contraste.

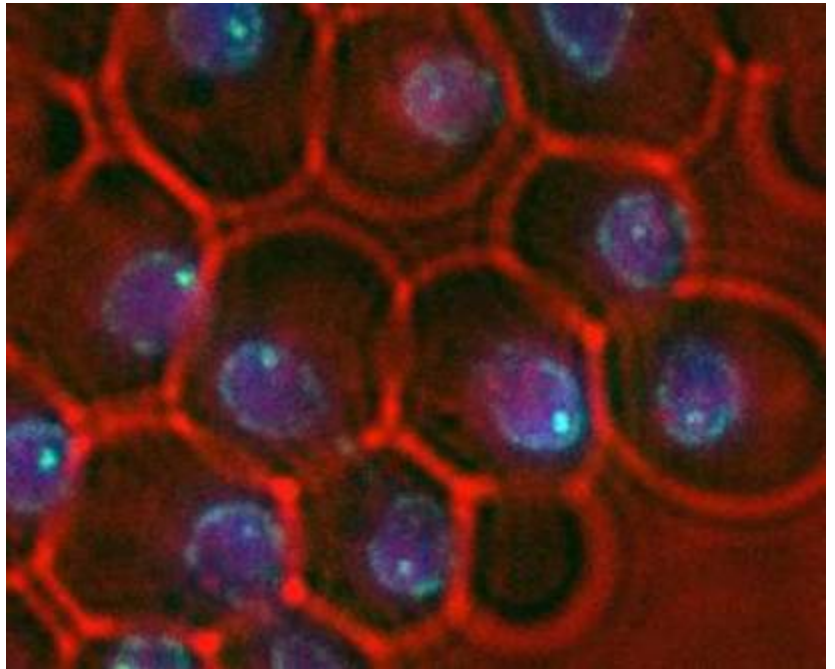
Escreva a rotina `rescale()` que muda o valor de pixel da imagem  $f(x, y)$  para:

$$g(x,y) = \alpha (f(x,y) - \beta)$$

$$\beta = \min(f(x,y))$$

$$\alpha = 255 / (\max(f(x,y)) - \min(f(x,y)))$$

Reescale  $g(x, y)$  a  $[0,255]$ .



Aplice a rotina à imagem de `microtubules.tif` chamando o "Rescale" de plugin. Observe o histograma.

2.3. Aplicação: Satura uma imagem médica.

Escreva a rotina `saturate()` que põe pixels a 10000 quando o valor de entrada é maior a 10000. Então chame a rotina `rescale()`.

Aplice a rotina em `HRCT.tif` chamando o plugin "Saturate". Observe o histograma.

3. A Segmentação por Limiar: "thresholding".

3,1 Contando partículas.

Usar o comando de "Limiar" de ImageJ, tenta segmentar as imagens: `yeast1`, `yeast2`, `yeast3` e `yeast4` para obter 25 objetos no analisador de partículas de ImageJ (Analisa -> Analisa Partículas). Se não for possível achar um limiar adequado, explique por que no arquivo de relatório.doc.

4. Projeção de "Z-Stack".

4.1. Projeção de máxima intensidade.

Escreva a rotina `zprojectMaximum()` que gera a imagem máxima de projeção de intensidade de uma pilha de imagens.

Aplique a rotina ao `yeard_stack.tif` por chamar o plugin ZMIP.

#### 4.2. Projeção de intensidade média.

Escreva a rotina `zprojectMean()` que gera a imagem média de uma pilha de imagens.

Aplique a rotina ao `yeard_stack.tif` por chamar o plugin ZMean.

#### 4.3. Apresentação de imagens de z-stack.

Usar dois plugins previamente, o comando "Brilho & Contraste", o comando "Image Calculator" → "Color Merge" do ImageJ na tentativa de obter uma imagem composta de cor onde claramente podemos distinguir a célula de fermento, o núcleo de GFP-tagged e o telomere de GFP-tagged (mancha). A fonte é uma z-stack de imagens fluorescentes chamada `yeast_stack.tif` e a imagem de fase é `yeast_phase.tif`. Na imagem acima está um exemplo de que nós podemos esperar. Coloque a imagem resultante no arquivo `relatório.doc`.