

# Exercício 1

- Crie uma aplicação que faz a leitura de um vetor de inteiros
- Após o usuário informar todos os números, uma thread deve ser criada para ordenar o vetor e exibir o vetor original e o vetor ordenado
- O usuário pode informar um novo vetor, ou seja, há uma estrutura de repetição no método `main()`

# Exercício 2

- Crie uma aplicação Hello World que inicie pelo menos 2 threads
- Cada thread deve imprimir na tela alguma mensagem que a identifique
  - Exemplo: Olá, sou a Thread número x!
- A mensagem deve ser exibida dentro de uma estrutura de repetição
- O programa contendo o `main()` deve informar
  - O horário que todas as threads terminaram
  - Qual foi a primeira thread a concluir sua execução
- Experimente os métodos ensinados em sala de aula para verificar o que acontece com a execução de cada thread

# Exercício 3

- Crie um programa para permitir que o usuário informe uma pasta contendo imagens e o programa as converta em PB
- Use as classes `BufferedImage` e `ImageIO`
  - Use o método `ImageIO.read(<file>)` para carregar a imagem
- Salve a imagem convertida no HD
  - `ImageIO.write(bufferedImage, "<extensão>", <newFile>);`
- Experimente utilizar um `programa monothread` e uma `versão multithread` com `uma thread para cada imagem` a ser processada
- A conversão da imagem pode ser feita usando o seguinte código

```
Color pixelColor = new Color(bufferedImage.getRGB(x, y));  
  
int blackAndWhiteNumber = (int) (pixelColor.getRed() * 0.3 + pixelColor.getGreen() * 0.59 + pixelColor.getBlue() * 0.11);  
  
Color grayscalePixel = new Color(blackAndWhiteNumber, blackAndWhiteNumber, blackAndWhiteNumber);  
  
bufferedImage.setRGB(x, y, grayscalePixel.getRGB());
```