**Relatório do LAB 3 - Sistemas Lineares**

Isaac de Lyra Junior - 01D - 20170117907

Túlio José Costa da Silva - 01D - 20170138326

Vilson Rodrigues Camara Neto - 01D - 20170138110

**Resumo**

Na terceira prática no laboratório fizemos 2 experimentos, um que visa criptografar uma imagem, e outro que descriptografa uma matriz. No primeiro, usamos técnicas de decomposição LU para dividir a imagem ao meio. No segundo desincriptografamos uma matriz usando os métodos de Jacobi e Gauss-Seidel, afim de descobrir a mensagem nela escondida. No fim, discutimos a eficiência dos métodos.

**Introdução**

No terceiro laboratório de Computação Numérica, fizemos 2 experimentos. O primeiro deles partiu de uma selfie dos integrantes do grupo, no formato quadrado, em seguida foi reduzida a resolução para 540x540 pixel’s, depois convertida em tons de cinza. Em seguida, visando utilizar conhecimentos de criptografia de forma de dificultar a interceptação por crackers. Dividimos a imagem em 2 usando a decomposição em L e U, método de Alan Turing que é uma forma de fatoração de uma matriz não singular como produto de uma matriz triangular inferior (lower) e uma matriz triangular superior (upper). Ao receber as matrizes L e U, basta multiplicar as mesmas e terá a imagem original.

Já o segundo experimento consistiu em descriptografar uma mensagem criptografada por uma matriz de criptografia. Para isso foram usados os métodos de Jacobi e o de Gauss-Seidel, ambos são métodos iterativos e funcionam de forma semelhante, o de Jacobi resolve o membro esquerdo da expressão em ordem a ao usar o método resultante da iteração anterior no membro direito. Já o de Gauss-Seidel utiliza valores encontrados na atual iteração, tendendo a ser mais eficiente na maioria dos casos. Em seguida, vamos comparar os resultados obtidos e discuti-los.

**Desenvolvimento/Resultados**

Parte 1 - Criptografia da imagem



Figura 1: selfie dos componentes do grupo na proporção original



Figura 2: imagem cortada na proporção 540x540 pixels



Figura 3: imagem em tons de cinza na proporção 540x540 pixels e suavemente borrada

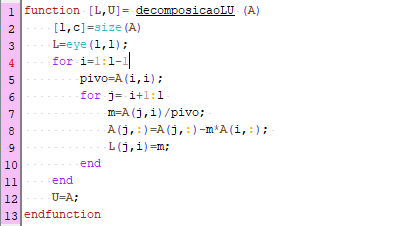


Figura 4: código utilizado na decomposição LU

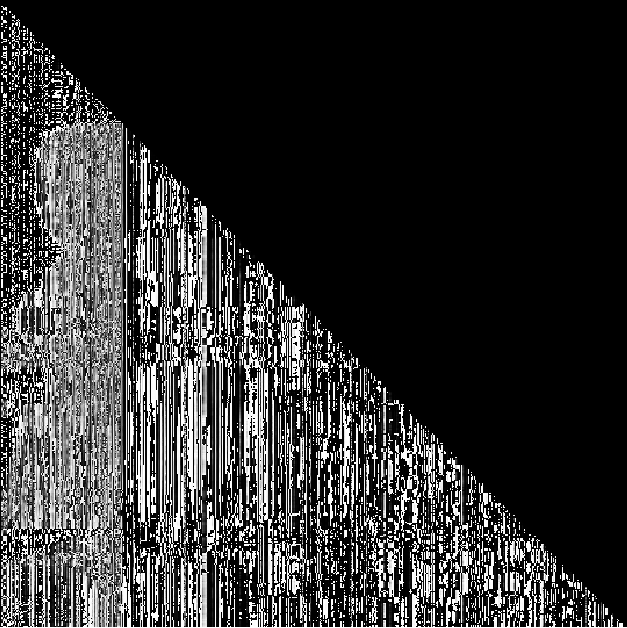


Figura 5: imagem plotada da matriz resultante L

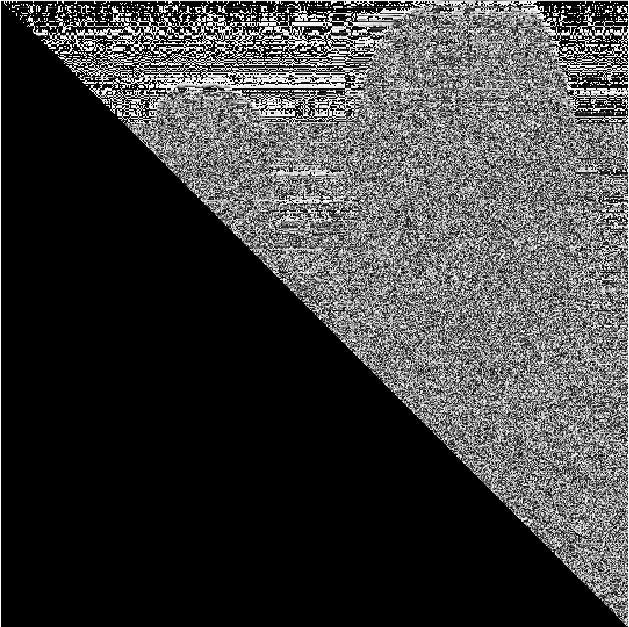


Figura 6: imagem plotada da matriz resultante U



Figura 7: foto do grupo reconstruída após plotada a multiplicação de matrizes L\*U



Figura 8: print do terminal e das chamadas para a realização do Experimento 3 - Parte 1

Parte 2 - Descriptografando mensagem

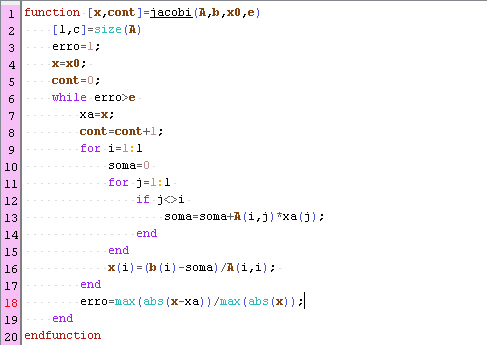


Figura 4: código função do método de Jacobi

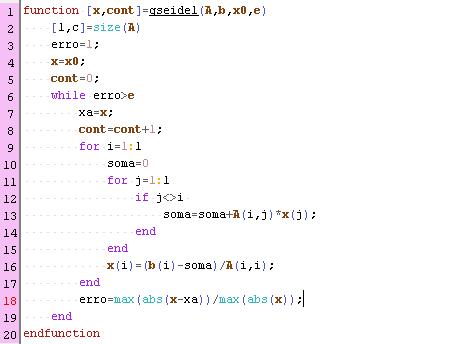


Figura 5: código da função do método de Gauss-Seidel

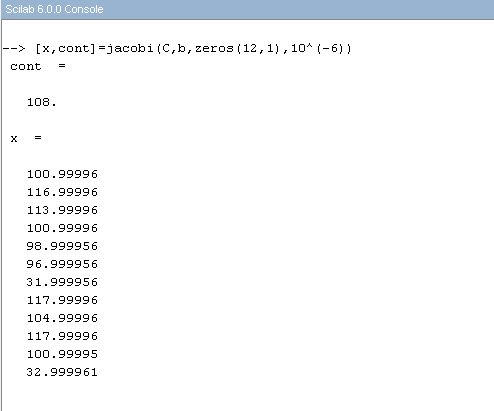


Figura 6: chamada da função do método de Jacobi e resultado

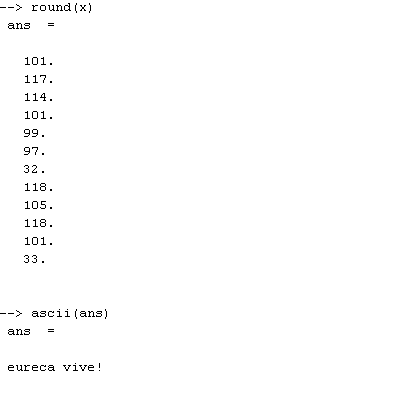


Figura 7: mensagem que estava criptografada achada pelo método de jacobi após ajustes

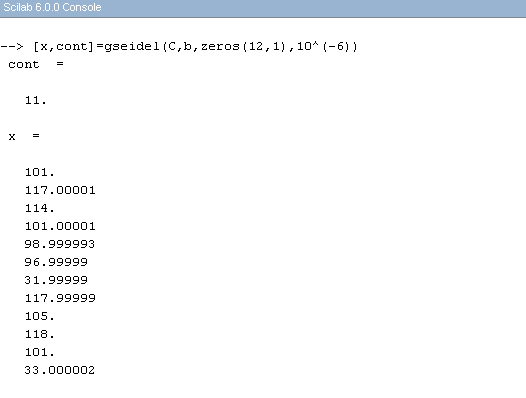


Figura 8: chamada da função do método de Gauss-Seidel e resultados

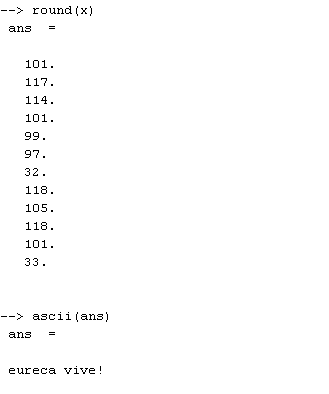


Figura 9: mensagem que estava criptografada achada pelo método de Gauss-Seidel após ajustes

**Discussões/Conclusões**

No primeiro experimento, o método de decomposição LU se mostrou eficiente, conseguindo separar a imagem em 2 matrizes e depois conseguindo reverter o processo e construir a imagem igual a original.

No segundo experimento, ambos os métodos conseguiram chegar na solução do sistema esperada, e também conseguiram desencriptografar a matriz C e achar a frase que foi inserida. O método de Jacobi conseguiu achar a frase com 108 iterações, e o de Gauss-Seidel conseguiu em 11 iterações. A diferença entre os métodos é somente que o Método de Gauss-Seidel utiliza da solução do x da linha anterior para calcular o novo x da linha atual, dessa forma converge mais rapidamente nesses casos para o vetor solução esperado.

**Referências bibliográficas**

Google Analytics**. Decomposição LU**. Disponivel em: <https://pt.wikipedia.org/wiki/Decomposi%C3%A7%C3%A3o\_LU> Acesso em: 14 abr. 2019.

# Google Analytics**. Método de Jacobi.** Disponivel em: <https://pt.wikipedia.org/wiki/M%C3%A9todo\_de\_Jacobi> Acesso em: 14 abr. 2019.

# 