

**Relatório do mini projeto de LP1**

**CifreX**

Grupo = ZUCAS\_2.0

Professor Teórico: Pedro Serra

Professor Prático: Guilherme Sanches

2020

**Índice**

Cifra de Substituição:

* Descrição da solução..............................................................
* Fluxograma.............................................................................

Decodificação da Cifra de Substituição

* Descrição da solução..............................................................
* Fluxograma.............................................................................

Cifra de Transposição

* Descrição da solução..............................................................
* Fluxograma.............................................................................

Decodificação da Cifra de Transposição

* Descrição da solução..............................................................
* Fluxograma.............................................................................

Matéria aprendida...........................................................................

Conclusão........................................................................................

Referências.....................................................................................

**Cifra de Transposição**

Para encriptar o texto passado para a função cifraDeTranposicao() dividimos o código em três partes:

1. Identificamos exatamente quantos caracteres possui o texto passado.
2. Passamos esse texto para uma matriz
3. Passamos os caracteres que estão na matriz para uma variável mas em uma ordem diferente.

Agora vou explicar em mais detalhes cada parte citada acima.

Na primeira parte o objetivo é descobrir quantos caracteres há no texto passado para que assim possamos descobrir a quantidade de linhas que a matriz terá tendo em conta o número de colunas passado.

Com um ciclo percorremos o texto passado a procura do caractere terminador, ‘\0’, assim que o encontramos, guardamos em uma variável a posição em que ele esta menos um, e assim obtivemos a quantidade de caracteres do texto.

Com esse número em mãos dividimos ele pelo número de colunas passados para a função, assim descobrimos quantas linhas serão necessárias.

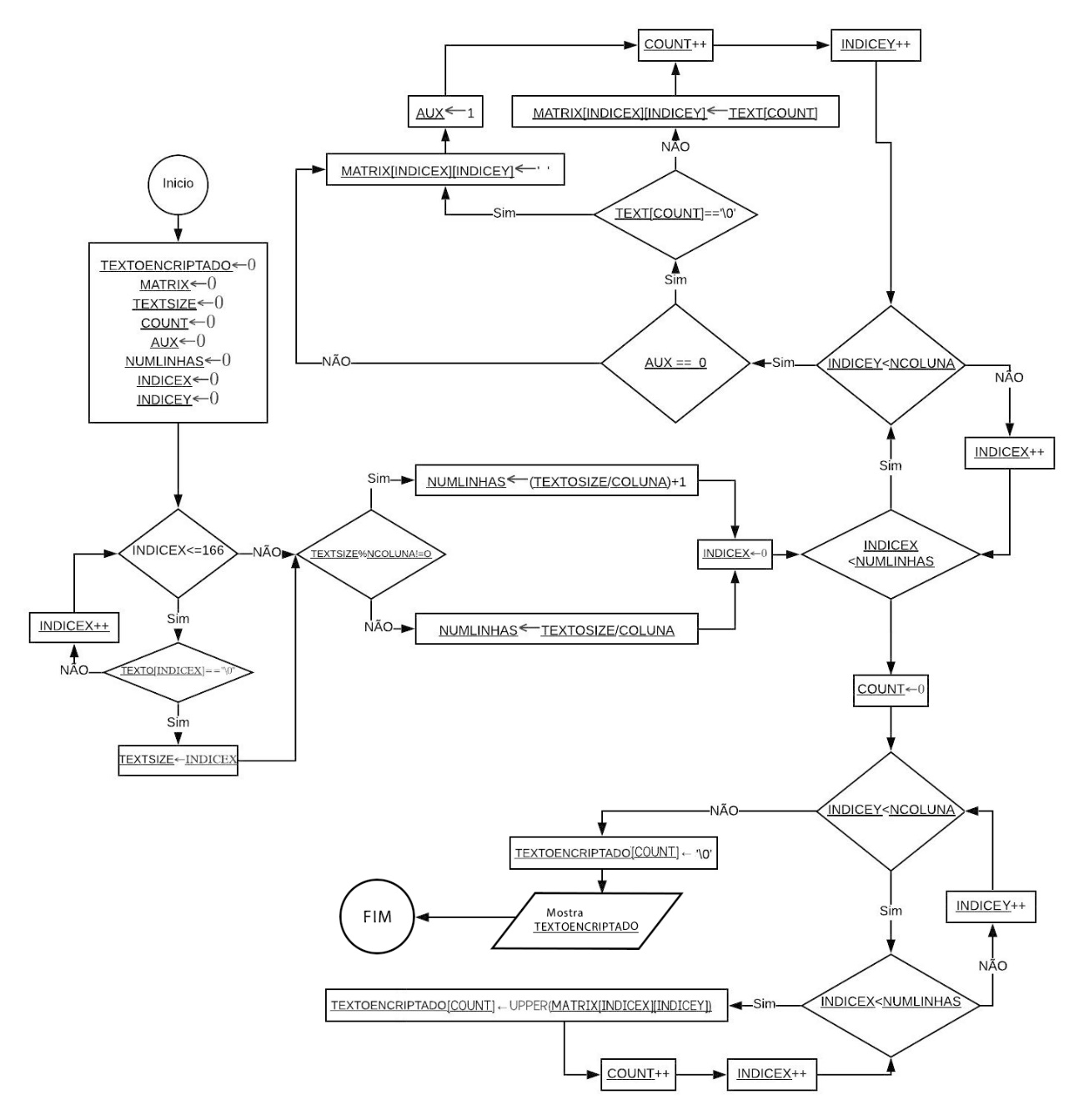
Agora usamos dois ciclos “for” um dentro do outro para percorrer a matriz. Um percorrendo as linhas outro percorrendo as colunas. assim vamos preenchendo-a com os caracteres do texto, por isso utilizamos uma variável (count) para ir incrementando as posições do texto que já havíamos copiado para dentro da matriz. Fazemos essa passagem sempre em busca do caractere ‘\0’, e assim que o encontramos preenchemos o resto da linha, se houver ainda espaços sobrando na linha da matriz, com um espaço.

Agora que preenchemos a matriz com o texto basta ler os caracteres de uma maneira diferente. Em vez de percorrer a matriz linha por linha, como fizemos para montar a matriz, vamos agora percorrer em coluna, ou seja, vamos ir lendo verticalmente a matriz.

Para fazer isso utilizamos os mesmos ciclos, um dentro do outro e vamos passando cada caractere para uma variável.

Assim que acabamos, acrescentamos o ‘\0’ no final desse texto criptografado e mostramos para o usuário.

Abaixo o fluxograma da função que faz cifra de transposição.

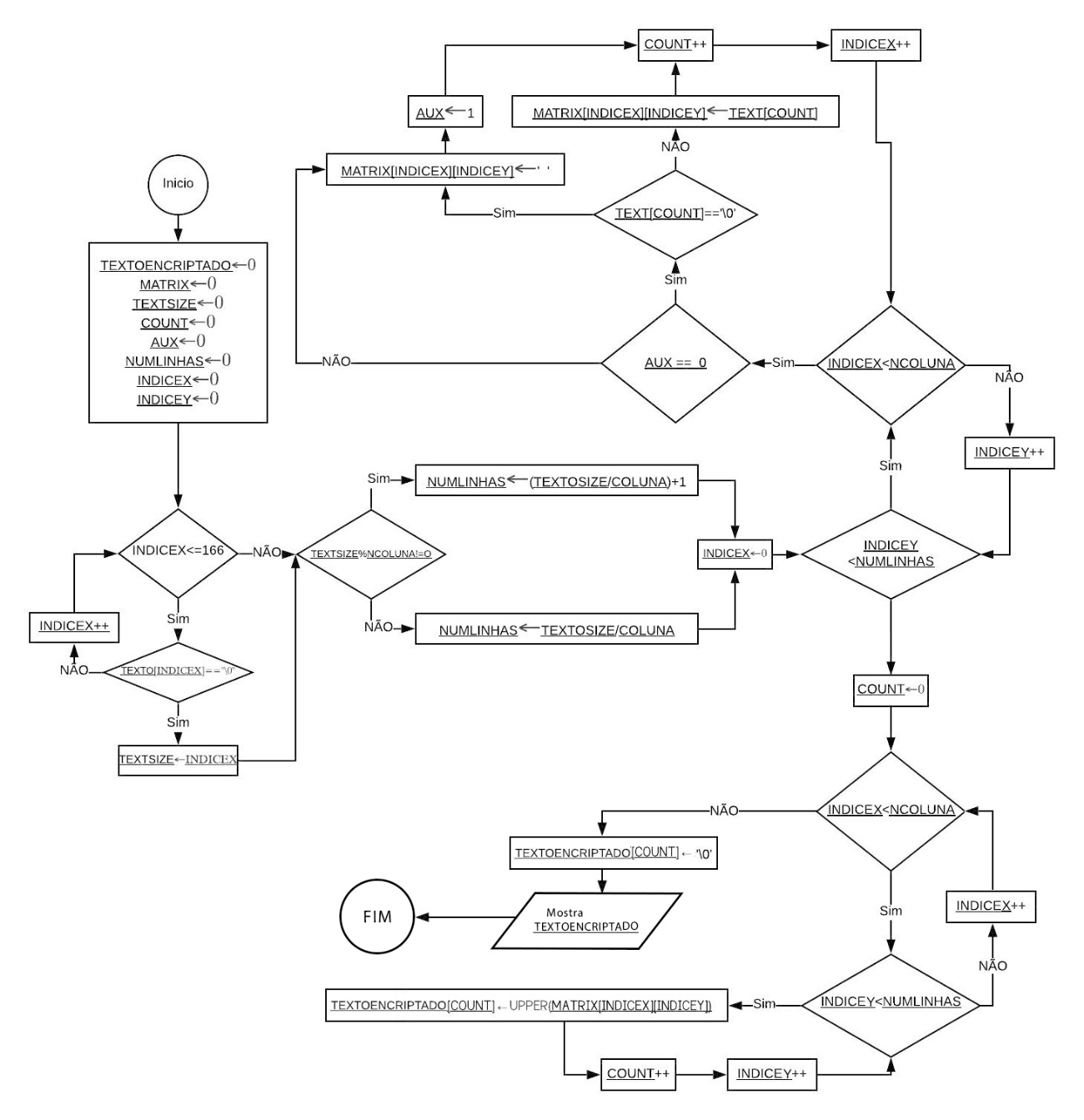


**Decodificar a Cifra de Transposição**

A decodificação da cifra de transposição é muito semelhante a parte de encriptação em questões de código. De uma maneira simples precisamos fazer as mesmas coisas que tínhamos feito, mas alterando onde era o número de coluna para número de linhas e vice versa.

Isso fara com que montemos o texto da maneira reversa ao que havíamos feito na parte de codificação, logo vamos decodificar o texto passado.

Abaixo o fluxograma da função que decodifica a cifra de transposição.



**Matéria aprendida**

Com o mini projeto de Linguagem de Programação I – CifreX nos deparamos com diversos Warnings e Errors, o que nos fez melhorar muito nosso código em C, tanto em qualidade de código, quanto velocidade e mesmo lógica. Citando algumas matérias, mais em específico, que conseguimos treinar, estão:

1. Mexer com ponteiros e referências.
2. O uso das matrizes.
3. A manipulação de strings.
4. A utilização de ciclos como for, while e do while.
5. O uso de funções
6. O uso de expressões regulares no scanf

**Conclusão**

O mini projeto nos proporcionou grandes aprendizados em C e nos deu a oportunidade de trabalhar em códigos que necessitam ser mais bem elaborados que os das fichas práticas. Com isso nos deparamos com muitos obstáculos que foram servindo de aprendizado sempre que os ultrapassamos.

Assim como tivemos que nos propor, pela primeira vez no curso, a realizar um relatório sobre um código que havíamos escrito, de maneira a não mostrar o código, e sim, interpretar como havíamos solucionado o problema e explicar em um texto.