

# Relatório T3 - Redes I

Adolfo Tognetti - GRR20152278  
Luiza Wille - GRR20141014

Novembro 2017

## Instruções para execução:

Para comprimir um arquivo execute o comando:

```
python main.py -c "arquivoEntrada.txt" "arquivoComprimido"
```

Para descomprimir um arquivo execute o comando:

```
python main.py -x "arquivoComprimido" "arquivoSaida.txt"
```

## Detalhes da implementação do compressor LZW:

O trabalho foi implementado na linguagem Python por conta de sua facilidade em lidar com strings e também por conter a estrutura de dados "Dictionary", que foi usada no projeto para construir a tabela ASCII estendida.

Para comprimir um arquivo de texto primeiro iniciamos a tabela ASCII de tamanho 256 utilizando uma estrutura de dicionário e então executamos o algoritmo. Ao final, teremos um vetor com a string comprimida que é escrita em um arquivo de saída binário.

No processo de descompressão entramos com o arquivo binário e construímos a mesma tabela inicial. Conforme o texto é descomprimido a tabela estendida é montada da mesma forma em que foi feita no processo de compressão. Na maioria dos casos o código lido já vai estar presente na tabela, porém quando a substring começar e terminar com o mesmo caractere, por exemplo "aba", estaremos em um caso especial onde a substring ainda não está na tabela, então devemos concatenar o primeiro caractere dela no fim, como em "ab"+"a" = "aba", e então a adicionamos na tabela. Quando não há mais nada a ser lido do arquivo de entrada o processo termina e o texto descomprimido é escrito em um arquivo de texto.