

# **Relatório referente ao trabalhos 2**

**Artur Temporal Coelho<sup>1</sup>, Iago Mello Floriano<sup>1</sup>, Tiago Serique Valadares<sup>1</sup>**

<sup>1</sup>Universidade Federal do Paraná, Curitiba, Brazil

{atc19, imf19, tsv19}@inf.ufpr.br

## **1. A Linguagem**

A linguagem escolhida para desenvolver os trabalhos foi Python 3, por ser fácil de utilizar seus recursos e tornar a implementação mais rápida e simples de ser desenvolvida.

## **2. O Algoritmo para detecção de erros**

O algoritmo escolhido foi o CRC-8 com polinômio WCDMA (0x9B). É um algoritmo usado geralmente em redes digitais e dispositivos de armazenamento a fim de detectar mudanças acidentais em dados digitais.

O algoritmo funciona da seguinte maneira, para cada byte dos dados do pacote, ocorre uma reflexão nesse byte e é feito um XOR com o valor de CRC atual, este valor gerado é usado em uma tabela com os valores de CRC-8 para o polinômio 0x9B pré-calculados e armazenados em memória, o valor recebido da matriz será o novo valor do CRC para a próxima iteração. Esse processo é realizado para todos os bytes do campo dados do pacote. Ao finalizar esse processo, o último valor CRC é refletido e retornado para ser usado como validador no pacote.

A validação do CRC é feita comparando o valor CRC recebido junto ao pacote com o CRC calculado a partir dos dados recebidos do pacote, se o valor der diferente, significa que há algum erro no campo de dados do pacote.

O CRC foi implementado no trabalho 1 como parte obrigatório para o funcionamento do trabalho.

## **3. O Algoritmo para correção de erros**

Foi escolhido o multidimensional parity-check para a apresentação sobre algoritmos de correção de erros. O algoritmo opera organizando a mensagem em uma grade multidimensional e calculando um dígito de paridade para cada linha e coluna. Em geral, um esquema de paridade n-dimensional pode corrigir  $n/2$  erros.