

TRABALHO PRÁTICO 0:

Cálculo de CRC de um arquivo

Sandro Miccoli - 2009052409 - smiccoli@dcc.ufmg.br
Leandro Duarte - -

¹Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

11 de outubro de 2012

***Resumo.** Esse relatório descreve como foi implementado o algoritmo de detecção de erros conhecido como Cyclic Redundancy Check (CRC). Neste trabalho específico iremos utilizar polinômios geradores de 8 e 16 bits.*

1. INTRODUÇÃO

O CRC é uma técnica utilizada para detecção de erros de transmissão de dados digitais. As mensagens a serem transmitidas são tratadas como códigos polinomiais, sendo representada por uma série de binários.

A cada bloco de dado transmitido é anexada um valor de checagem (*check value*), que é baseado no resto de uma divisão polinomial entre o conteúdo dos dados e um polinômio gerador. Quando esses dados são recebidos, é feito o mesmo cálculo e, caso os valores de checagem não coincidam, pode-se inferir que ocorreu um erro na propagação desses dados [Wikipedia].

2. MODELAGEM

”Uma descrição sucinta dos algoritmos e dos TAD, das principais funções e procedimentos e decisões de implementação.”

3. SOLUÇÃO PROPOSTA

4. IMPLEMENTAÇÃO

4.1. Código

4.1.1. Arquivos .c

- **crc.c** Blablablabla

4.1.2. Arquivos .h

- **crc.h** Bla bla bla

4.2. Compilação

O programa deve ser compilado através do compilador GCC através de um makefile ou do seguinte comando:

```
gcc -Wall -Lsrc src/crc.c -o crc
```

4.3. Execução

A execução do programa tem como parâmetros:

- Um arquivo de entrada (texto ou binário)
- Um índice do polinômio gerador, que pode ser 0 ou 1, para polinômios de 8 e 16 bits, respectivamente.

O comando para a execução do programa é da forma:

```
./crc <arquivo binario de entrada> <índice do polinômio - 0 ou 1>
```

5. AVALIAÇÃO EXPERIMENTAL

5.1. Resultado

6. CONCLUSÃO

Referências

Wikipedia. Cyclic redundancy check. http://http://en.wikipedia.org/wiki/Cyclic_redundancy_check.