Ignacio Morales Chang
Johnny Agüero Sandi
ExtraClase 3 – Algoritmos y Estructuras de Datos II
Semestre I – 2021

TEC | Tecnológico de Costa Rica

Comprobación de paridad

Bit de paridad no significa más que un bit adicional añadido a los datos en el transmisor antes de transmitir los datos. Antes de agregar el bit de paridad, se calcula el número de 1 o ceros en los datos. Basado en este cálculo de datos se añade un bit extra a la información real / datos. La adición de bits de paridad a los datos dará como resultado el cambio de tamaño de la cadena de datos. Esto significa que, si tenemos datos de 8 bits, después de agregar un bit de paridad a la cadena binaria de datos, se convertirá en una cadena de datos binarios de 9 bits.

Hay dos tipos de bits de paridad en la detección de errores, son

Paridad uniforme: Si los datos tienen un número par de 1, el bit de paridad es 0. Por ejemplo: data es 10000001 -> bit de paridad 0. Número impar de 1, el bit de paridad es 1. Por ejemplo: los datos son 10010001 -> el bit de paridad 1

Paridad impar: Si los datos tienen un número impar de 1, el bit de paridad es 0. Por ejemplo: los datos son 10011101 -> el bit de paridad 0. Número par de 1, el bit de paridad es 1. Por ejemplo: los datos son 10010101 -> el bit de paridad 1

Si los bits de paridad enviados en el transmisor y los bits de paridad recibidos en el receptor no son iguales, se detecta un error.

Cyclic Redundancy Check (CRC)

Un código cíclico es un código de bloque lineal (n, k) con la propiedad de que cada cambio cíclico de una palabra de código da como resultado otra palabra de código. Donde k representa la longitud del mensaje en el transmisor. n es la longitud total del mensaje después de agregar bits de comprobación (datos reales y los bits de comprobación). n, k es el número de bits de comprobación.

Los códigos utilizados para la comprobación de redundancia cíclica allí por detección de errores se conocen como códigos CRC (códigos de verificación de redundancia cíclica). Los códigos de comprobación de redundancia cíclica son códigos cíclicos acortados. Estos tipos de códigos se utilizan para la detección y codificación de errores.

Generación de código CRC

En función del número deseado de comprobaciones de bits, agregaremos algunos ceros (0) a los datos reales. Esta nueva secuencia de datos binarios se divide por una nueva palabra de longitud n + 1, donde n es el número de bits de verificación que se van a agregar. El recordatorio obtenido como resultado de esta división módulo 2- se añade a la secuencia de bits de dividendo para formar el código cíclico. La palabra de código generada es completamente divisible por el divisor que se utiliza en la generación de código. Esto se transmite a través del transmisor.

En el lado del receptor, dividimos la palabra de código recibida con el mismo divisor para obtener la palabra de código real. Para una recepción de datos sin errores, el recordatorio es 0. Si el recordatorio es distinto de cero, eso significa que hay un error en el código recibido / secuencia de datos.

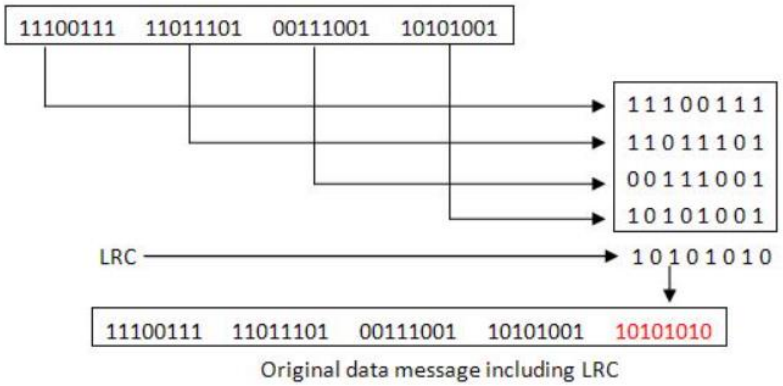
Longitudinal Redundancy Check

En verificación de redundancia longitudinal, un bloque de bits será ordenado en una table y se calculará la paridad de bits para cada columna por separada. El set de esta paridad de bits será enviado cuanto con el dato original de bits.

Verificación de redundancia longitudinal hace cálculos de paridad bit por bit a medida que calculamos la paridad de cada columna de manera individual.

Este método puede detectar fácilmente errores de ráfaga y errores de bit individuales y falla en detectar 2 errores de bit que ocurren en la misma rebanada vertical.

A continuación, se muestra un ejemplo de LRC.



Github: https://github.com/johnnyzaet08/TareaExtra_DatosII_3_4