

Übung Fehlererkennung

Übung 1: Parität

1. Schreibe eine Funktion in einer dir geläufigen Programmiersprache, die das Paritätsbit einer beliebigen binären Zahl überprüft. Das Paritätsbit ist das letzte Bit der zu prüfenden Zahl.
2. Schreibe eine Funktion, die für eine beliebige binäre Zahl das Paritätsbit (gerade Parität) berechnet. Prüfe diese Funktion mit der vorgängig erstellen Funktion.
3. Überprüfe, welche Fehler mit einem Paritätsbit erkannt werden können und welche nicht. Halte deine Erkenntnisse fest.

Übung 2: Checksumme

1. Schreibe eine Funktion, die für ein beliebig langes Array von binären 16bit Zahlen die Checksumme berechnet. Verwende dafür die bei TCP verwendete Einerkomplement-Addition:

```
def ones_comp_add16(num1,num2):  
    MOD = 1 << 16  
    result = num1 + num2  
    return result if result < MOD else (result+1) % MOD
```

2. Extrahiere mit Wireshark die nötigen Daten aus einem TCP Paket und überprüfe damit, ob du die Prüfsumme korrekt berechnen kannst.
3. Überprüfe, welche Fehler mit einer Checksumme erkannt werden können und welche nicht. Halte deine Erkenntnisse fest.