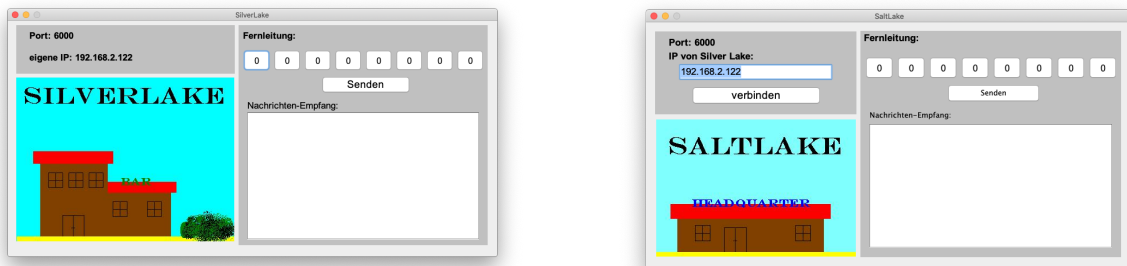


Rechnernetze Prüfverfahren

Mit den Erfahrungen aus *New Town* und *Gold City* ziehen die beiden Städte *Silver Lake* und *Salt Lake* ebenfalls eine Leitung, die auf der Bitübertragung aufsetzt. Über eine Leitung wird eine Reihe von 0 und 1 übertragen.



Aufgabe 1

Arbeitet in denselben Zweiergruppen, wie in der letzten Stunde und schaut euch erneut eure Zeichencodierung an. Es können immer acht Bit pro Nachricht übermittelt werden (siehe Abbildungen oben). Passt eure Codierung so an, dass sie mit dieser Art der Datenübertragung nutzbar ist.

Aufgabe 2

Einer von euch startet das Programm *SilverLake*, der andere von euch startet an irgendeinem anderen Rechner das Programm *SaltLake* aus dem Tauschverzeichnis. Kommuniziert miteinander. Teilt dem anderen zum Beispiel mit, was euer momentaner Lieblingsfilm ist oder auf welches Gericht ihr im Moment Appetit hättet.

Setzt euch nach etwa 10 Minuten „Unterhaltung“ wieder zusammen.

Aufgabe 3

Hat die Kommunikation einwandfrei funktioniert oder gab es Verständigungsschwierigkeiten? Lassen sich die Verständigungsschwierigkeiten auch auf technische Probleme zurückführen? Haltet eure Erkenntnisse in Stichpunkten fest.

Fehler bei der Datenübertragung können mit unterschiedlichen Prüfverfahren erkannt werden. Drei wichtige Verfahren sind die folgenden:

- *Paritätsbit*: Dem eigentlichen Datenblock wird ein sogenanntes *Paritätsbit* angehängt. Sind im Datenblock eine gerade Anzahl von Einsen vorhanden, so wird eine Null angehängt, andernfalls eine Eins.
- *Prüfsumme*: Zuerst wird die Quersumme des Datenblocks berechnet. Von dieser Summe wird dann der Rest modulo n berechnet und dem eigentlichen Datenblock angehängt.
Hinweis: Bei $n = 2$ entspricht dies dem Paritätsbit-Verfahren. Warum?
- *gewichtete Prüfsumme*: Die einzelnen Bits des Datenblocks werden mit einer individuellen Gewichtung (z. B. mit der Wichtungsfolge $1, 2, 3, 4, \dots, m$) in die Quersumme eingerechnet. Auch hier wird der Rest modulo n an den Datenblock angehängt.

Aufgabe 4

Wende die drei Prüfverfahren jeweils auf die Zahlenfolgen an:

a) 11001

b) 11000

c) 10101

d) 00010

Warum ist es bei den Prüfsummenverfahren sinnvoll, für n eine Zweierpotenz zu wählen?

Aufgabe 5

Häufige Fehlerquellen sind: Kippen eines Bits und Vertauschen zweier Bits. Zeige jeweils an einem Beispiel, welches der drei Prüfverfahren den ersten bzw. zweiten Fehler erkennt bzw. nicht erkennt.

Aufgabe 6

Welche der Sendungen sind wahrscheinlich fehlerfrei?

Prüfbit:	a) 11001 1	b) 10111 1	c) 00000 0	d) 11111 1	e) 00101 1
Quersumme ($n = 4$):	a) 11001 11	b) 10111 01	c) 00000 00	d) 11111 01	e) 00101 10
gew. Quers. ($n = 4$):	a) 11001 00	b) 10111 10	c) 00000 10	d) 11111 11	e) 00101 00