

## ÜBERBLICK

- Stellenwertsysteme
- ASCII-Code, d.h. Speicherung von Text durch Nullen und Einsen
- Logische Ausdrücke/Schaltungen
- NP-Klassen: disjunktive Normalform
- 4-Bit-Addierer (zur Addition zweier 4-stelliger Binärzahlen) mit Logism (wie er aussieht, aus welchen Bauteilen er besteht (Halbaddierer, Volladdierer, welche Wahrheitstabelle diese haben))
- Umsetzen logischer Ausdrücke als logische Schaltung mit Logisim.

Wenn das in der Klausur drankommt: Abgabe per `circ`-Datei (das ist die Logisim-Dateiart) oder per Screenshot der erstellten Schaltung (am besten inklusive sichtbarer, von Logisim erstellter Wahrheitstabelle).

### LERNZIELE ZUM THEMA “BITS AND BYTES” BZW. WAS MAN FÜR DIE PRÜFUNG KÖNNEN SOLLTE

- St. Galler Bahnhofsuhr ablesen können
- Zahlen zwischen verschiedenen Stellenwertsystemen umrechnen können (Binär-, Dezimal-, Fünfer- und Hexadezimalsystem) (vgl. Divisionsmethode (Algorithmus 1.1.6 im Skript) und Umwandlung von binär zu hexadezimal (Algorithmus 1.1.10))
- „Primarschulstoff“ in diesen Systemen (also Zählen, Kleines Einspluseins, Kleines Einmaleins, schriftliche Addition und Multiplikation)
- ASCII-Code: Folge von Binär- oder Hexadezimalzahlen in Folge von Zeichen umwandeln können (falls ASCII-Tabelle gegeben).
- Wissen, was ein Byte ist; was eine Kilobyte, Megabyte, Gigabyte, Terabyte ist.
- Kenntnis der logischen Verknüpfungen (AND, OR, NOT)
- Wahrheitstafeln zu logischen Ausdrücken angeben können
- NP-Klassen (also nicht Klasse 2pG): logischen Ausdruck zu Wahrheitstafel angeben können (intelligentes Raten oder per disjunktiver Normalform (siehe Beweis von Satz 1.4.8 im Skript))
- logischen Ausdruck als “Logisim”-Schaltung zeichnen (auch per Software) und umgekehrt logischen Ausdruck (bzw. logische Ausdrücke bei mehreren Outputs) zu gegebener “Logisim”-Schaltung angeben können..
- Wahrheitstabelle von Halbaddierer und Volladdierer sollten bekannt sein, ebenso Struktur eines 4-Bit-Addierers (vgl. Screenshots auf Informatik-Webseite)