

Zuerst zur Orientierung: Was wir gemacht haben – insbesondere, damit ihr die relevanten Blätter auf unserer Web-Site findet:

- Stellenwertsysteme (handschriftlich, 05. Januar)
- Logische Schaltungen (handschriftlich, 12. Januar)
- Wie Computer addieren: Arbeitsblatt `schaltlogik.pdf` vom 19. Januar + was wir mit Logisim gemacht haben (bis Anfang Februar)
- Clock arithmetic ~~und negative (Zahlen)~~ (handschriftlich, 9. bzw. 15. Februar)
- Codierung von Texten und Codierung von Bildern, d. h. das Arbeitsblatt `ascii-utf-8-bildcodierung-pruefziffern.pdf` (begonnen am 22. bzw. 23. Februar)
OHNE Seite 3 zu Prüfwerten
- KEIN PRÜFUNGSSTOFF, war nur zum besseren Verständnis und zur Verbindung mit Python gedacht: danach in Python rumprogrammiert, d. h. RGB-Farbwürfel als ppm-Datei in 2aLM, in 2dNP zusätzlich Rekursion, (farbiges) Sierpinski-Dreieck als SVG-Ausgabe.

LERNZIELE ZUM THEMA “BITS AND BYTES” BZW. WAS IHR FÜR DIE PRÜFUNG KÖNNEN SOLLTET

Hier nun, was ihr können solltet:

- Zahlen zwischen verschiedenen Stellenwertsystemen umrechnen, insbesondere Binär-, Dezimal- und Hexadezimalsystem
- einfache Rechnungen in diesen Systemen (also etwa $A+9$ im Hexadezimalsystem)
- Kenntnis der logischen Verknüpfungen (AND, OR, NOT)
- Wahrheitstabellen zu logischen Ausdrücken angeben
- logischen Ausdruck zu Wahrheitstabelle angeben
- logischen Ausdruck als “Logisim”-Schaltung zeichnen und umgekehrt logischen Ausdruck (bzw. logische Ausdrücke bei mehreren Outputs) zu gegebener “Logisim”-Schaltung angeben.
- Wahrheitstabelle von Halbaddierer und Volladdierer sollten bekannt sein, ebenso Struktur eines 4-Bit-Addierers
- Rechnen mit Uhrzeiten (insbesondere auf der 16-Stunden-Uhr). (was ist $13 + 9$? einfache Gleichungen $2x = 10$ (diese hat zwei Lösungen!) lösen, beispielsweise durch Ausprobieren aller Möglichkeiten)
(Wie man negative Zahlen auf dieser Uhr darstellen kann verlange ich nicht.)
- RGB-Modell für Farben: Die Codierung der $8 = 2^3$ Farben schwarz, rot, blau, grün, cyan, gelb, magenta und weiss sollte bekannt sein (jede der drei Intensitäten ist dabei 0 oder 255)
- ASCII-Codierung, d. h. Umwandlung von ASCII-Code in Zeichen und umgekehrt (falls dazu eine Aufgabe kommt, werde ich eine Tabelle bereitstellen)
- Ihr wisst, was Rastergrafik bzw. Vektorgrafik bedeutet und kennt somit auch den Unterschied. Ihr könnt entscheiden, für welche Bilder man was bevorzugt verwendet.
- einfache ppm-Dateien verstehen (wie die mit dem Haus auf dem Arbeitsblatt, wo ihr das Licht eingeschaltet habt), d. h. als Bild malen und umgekehrt.
- 2dNP: einfache svg-Dateien verstehen (Befehl `polyon`)