DMA

– Week 5 –

Work instructions

The topic of this week is mathematical logic and induction. Mathematical logic can be viewed as a formalization of well-known logical concepts. Note that in many cases, common sense is sufficient to construct a mathematical argument. However, when mathematical statements become increasingly complicated it may also be necessary to use a formal toolbox to evaluate the truth values of such statements. We will, e.g., introduce a notation suitable for describing what it means that an asymptotically positive sequence (a_n) is $O(1)$ as

$$\exists c > 0 \; \exists k > 0 \; \forall n \geq k \;\; a_n \leq c$$

Some of the concepts and methods that we will introduce will reappear in your later studies of machine architecture.

We will be reading KBR 2.1–2.2 very thoroughly. This is not easy material so we recommend that you read it at least two times: once before the lectures and once after.

On Tuesday and Friday we'll cover one of the most important topics in the math part of DMA: mathematical induction. This will be our preferred method to prove a sequence of statements such as

$$P(n) : \sum_{k=1}^{n} k = \frac{n(n+1)}{2}.$$

During this week we will see how to set up and execute a proof by induction.

Since mathematical induction is such an important element also for the algorithmic part of the course, we'll deviate a bit from our KBR textbook. We have prepared some brief notes on mathematical induction that should be an easier read than KBR and only contain what is needed for this week. The difference, however, is not that big so we recommend that you also skim Section 2.4 in KBR.

Assigned reading

- KBR 2.1–2.2. Read this carefully. This material will be useful for the Monday and Tuesday lectures.
- Notes for week 5 (on Absalon). You can also take a quick look at KBR Section 2.4. We will use this material in the second lecture on Tuesday and on Friday.

Lecture plan

Monday Oct. 2nd 09:15–10:00

Propositions, logical operations/connectives ($\wedge, \vee, \sim, \Rightarrow, \Leftrightarrow$), truth tables, tautologies, and absurdities.

Tuesday Oct. 3rd, 13:15-14:00

Predicates, quantifiers ($\forall$ and $\exists$), negation of statements with quantifiers.

Tuesday Oct. 3rd, 14:15–15:00

The principle of mathematical induction.

Friday Oct. 6th, 09:15–10:00

Strong induction. Examples.

Exercise plan

Monday Oct. 2nd, 10:15–12:00

- Solve KBR exercises 2.1.1, 2.1.2, 2.1.8, 2.1.27, 2.1.28.
- Check De-Morgan's laws by computing and comparing the truth tables of the left-hand-side and right-hand-side in each of the following

  1. $\sim (p \vee q) \equiv (\sim p) \wedge (\sim q)$
  2. $\sim (p \wedge q) \equiv (\sim p) \vee (\sim q)$

- Solve KBR 2.1.37, 2.1.38.

- Solve KBR exercises 2.2.10, 2.2.11, 2.2.13, 2.2.15.
- Let `xor` be a logical connective with the following truth table:

P	Q	P xor Q
T	T	F
T	F	T
F	T	T
F	F	F

  Find an equivalent expression for

$$P \texttt{ xor } Q$$

  using only $\wedge$(and), $\vee$(or), and $\sim$(not). You can use P and Q any number of times and indicate the order of the operations using parenthesis. Verify your answers by computing the truth table of your expression. (See Example 6 in KBR 2.1 for an example.)

Tuesday Oct. 3rd, 15:15–17:00

- Solve KBR exercises 2.1.15, 2.1.16, 2.1.18
- Let $f, g : \mathbb{R}^+ \to \mathbb{R}$ be asymptotically positive functions. We can express the definition of "$f(x)$ is $O(g(x))$" from Week 3 using logical connectives and quantifiers as

$$\exists c > 0 \; \exists x_0 \in \mathbb{R}^+ \; \forall x \geq x_0 \; f(x) \leq cg(x) \tag{1}$$

  Recall from Week 3 that we defined "f is $o(g)$" as "for any constant $c > 0$ we can find $x_0 \in \mathbb{R}^+$ such that

$$f(x) < cg(x)$$

  for all $x \geq x_0$."

  1. Express the above definition of "$f(x)$ is $o(g)$" using logical connectives and quantifiers.
  2. Write the negation of the proposition from the previous part and simplify it so that it does not contain the negation ($\sim$).
     Hint: Theorem 3 from KBR 2.2 can be helpful here.
  3. Write a sentence in English that corresponds to your statement from the previous part.

- Solve KBR exercises 2.2.6, 2.2.21.
- [*] Solve KBR 2.2.27.
- Solve KBR exercises 2.4.3, 2.4.4, and 2.4.8.

Friday Oct. 6th, 10:15-12:00

- If needed, the instructor explains loop invariants (KBR page 72). Instructor uses loop invariants to solve KBR 2.4.36.
- Solve KBR 2.4.35.
- Solve KBR exercises 2.4.6, 2.4.10, 2.4.16, 2.4.22, 2.4.27, 2.4.29. Solve any leftover problems from Monday or Tuesday.