IDMA 2025

– Ugeseddel 5 –

General Plan

We will start this week by talking a bit about **matrices** and how they can be useful in discrete mathematics. Most (or all) of you would also be taking a separate course on linear algebra, but for many (or all) of you it comes after this course, and so cannot help us with the material we need. Therefore, just to make sure we are on the same page, we will make sure to quickly cover the basics in a self-contained and elementary way.

We then come to the topic of **relations**. As we all know, relations are some of the most important topics in life, and that holds true also in mathematics, so we will focus on relations this week. Only, in mathematics the word "relation" has a very precise meaning, which might be slightly different from our every-day usage. In fact, we have already been using mathematical relations, and properties of them, in several previous lectures, and so it is high time that we pin down exactly what we mean.

Informally, relations between integers are things like *"a is less than b"* (which we usually denote $a < b$) or *"a divides b"* (denoted $a|b$). Other examples of relations are between people, such as *"A is a friend of B"* or *"A is a parent of B."*

Mathematically, relations can be viewed in different ways:

- as a logical predicate $P(a, b)$ regarding a pair of elements $a \in A$ and $b \in B$,
- as a subset of the *product set* $A \times B$,

and if A and B are finite sets:

- as a Boolean matrix,
- as a directed graph (digraph) when $A = B$.

It might seem a little extreme to have four different descriptions of the same concept. However, it is useful to be able to represent relations in different ways, as then we have the freedom to choose the description best suited for the task at hand.

During our treatment of relations, we will also make a detour to talk about **functions** to see that they are relations of a special form, and then discuss the concepts of *injective*, *surjective*, *bijective*, and *inverse* functions.

We will then talk about properties of relations such as *reflexivity*, *symmetry*, and *transitivity*. These three properties together define an important family of **equivalence relations**.

Continuing our discussion of relations, we will consider two important types: **order relations** and **trees**. The former is relevant in computer science because this type of relation is what we use when we sort data. The latter will lead us on to a more general discussion of **graphs**—which will be the topic of the last two weeks of lectures on the course.

Reading Instructions

- KBR Section 1.5 on matrices (Monday)
- KBR Chapter 4 except sections 4.6 and 4.8 (Wednesday morning)
- KBR Section 5.1 (Wednesday morning)
- KBR Sections 6.1–6.2 (Wedesday afternoon)
- KBR Sections 7.1 and 7.2 (Wednesday afternoon)
- KBR Section 8.1 (Wednesday afternoon)

Please note that there is quite a lot of material in Chapter 4, and a substantial part of it is introducing formal terminology for different notions. We will not be able to cover all of this in detail in class, so it is important (as is always the case, but even more so here) that you also read through the material carefully on your own.

Although we will not talk about it this week, parts of Section 7.5 will be covered in a week or two. The same goes for Section 7.3 with traversals of trees. In other words, it does not hurt reading these sections already now (and in any case they are part of your general computer science education).

Also, in case you want to read on after Section 8.1, then in Sections 8.2 and 8.3 you find a discussion of Eulerian and Hamiltonian cycles, which you might remember is what we started talking about in the very first lecture on the course. We probably will not have time to talk about these sections, though (or at least that is not in the plans as they are looking now).

Just to avoid confusion—in case you would start wondering why there is no mention of Section 7.5 in the course plan—the reason for this is that while we will cover this material, we will do so by reading CLRS instead.

Exercises

Note that as for previous weeks, there are quite a few exercises suggested below. If you feel that you understood what was covered in class and/or what you read in the textbook, and if the exercises seem straightforward, then it is fine to just do some of them (where I good idea would be to try to focus on the one that seem hardest to you). If some of the material seems harder, though, then it is a good idea to do more exercises until you really understand what is going on.

Monday material:

1. Solve KBR exercises 1.5.5, 1.5.9, 1.5.12, 1.5.16, and 1.5.21.
2. Solve KBR exercises 1.5.40, 1.5.42, 1.5.43.

Wednesday material:

1. Solve KBR exercises 4.1.5, 4.1.10
2. Solve KBR exercises 4.2.4, 4.2.9, 4.2.10, 4.2.23, 4.2.25
3. Solve KBR exercises 4.3.1–2, 4.3.4–8, 4.3.19
4. Solve KBR exercises 4.4.1, 4.4.13
5. Solve KBR exercises 4.5.3, 4.5.4, 4.5.8, 4.5.12
6. Solve KBR exercises 4.7.2, 4.7.7, 4.7.12, 4.7.16–17, 4.7.19
7. Solve KBR exercises 5.1.1, 5.1.11, 5.1.30
8. Solve KBR exercises 6.1.2, 6.1.6, 6.1.9, 6.1.13,
9. Solve KBR exercises 6.2.1, 6.2.2, 6.2.9, 6.2.10, 6.2.13–14
10. Solve KBR exercises 7.1.1–4, 7.1.9–13, 7.2.1–2, 7.1.24–26
11. Solve KBR exercises 8.1.1–2, 8.1.5, 8.1.17, 8.1.19, 8.1.20.

More advanced exercises:

1. Discuss the following among yourselves:
    - Write pseudocode that determines whether a relation given as a list of elements from $A \times B$ is

    - reflexive
    - irreflexive
    - symmetric
    - asymmetric
    - antisymmetric
  - Write pseudocode that does the same for a relation given as a Boolean matrix.

2. Solve KBR exercises 6.1.19, 6.1.20.

3. Solve KBR exercises 6.1.21, 6.1.22.

4. *(Partitioning a poset into linear orders)* Solve parts (e) and (d) of KBR 6.1.26–27

5. *(Proof exercise)* Solve KBR 6.1.28

6. Solve KBR exercises 6.2.17–18, 6.2.36

7. *(Proof exercise)* KBR 6.2.20

8. *(Proof exercises)* Solve KBR 7.1.29–30, 7.1.31