Plan

Introduce yourself to those of your classmates sitting closest to you and tell them a bit about yourself. We are guaranteed to be a population with very different backgrounds, different languages, different hometowns, different career plans, and so on. If you already know your neighbours, mingle with some of the other students.

In the first week, and certainly before the tutorial, make sure you read Chapter 1 of Levitin, so that all of the following questions make sense. We may cover some of the questions in a lecture, and some we may have to skip, but work on all of them in your own time.

In general, over the first few weeks, you may meet concepts that we have not yet covered systematically, such as "sorting", "undirected graph" or "adjacency matrix". Don't let that unsettle you. For now we just want to dip our toes in a variety of problems; later we will cover things in greater detail.

The exercises

1. Consider the usual (unsigned) binary representation of integers. For example, `10110010` represents 178, and `000011` represents 3.

   (a) If we call the bits in an n-bit word $x_{n-1}, x_{n-2}, \ldots, x_2, x_1, x_0$ (so x_0 is the *least significant* bit), which natural number is denoted by $x_{n-1}x_{n-2}\cdots x_2x_1x_0$?

   (b) Describe, in English, an algorithm for converting from binary to decimal notation.

   (c) Write the algorithm in (pseudo-) code.

   (d) Describe, in English, how to convert the decimal representation to binary.

2. Below are three (correct) formulas for calculating the area of a triangle with sides A, B, and C, of length a, b, and c, respectively.

   (a) $S = \sqrt{p(p-a)(p-b)(p-c)}$, where $p = (a+b+c)/2$

   (b) $S = \frac{1}{2}ab\sin\theta$, where θ is the angle between sides A and B

   (c) $S = \frac{1}{2}ah_A$, where h_A is the height to base A

   Which of these would you accept as an *algorithm*?

3. Consider the following problem: You are to design an algorithm to determine the best route for a subway passenger to take from one station to another in a city such as Kolkata or Tokyo.

   (a) Discuss ways of making the problem statement less vague. In particular, what is "best" supposed to mean?

   (b) How would you model this problem by a graph?

4. Consider the following map:
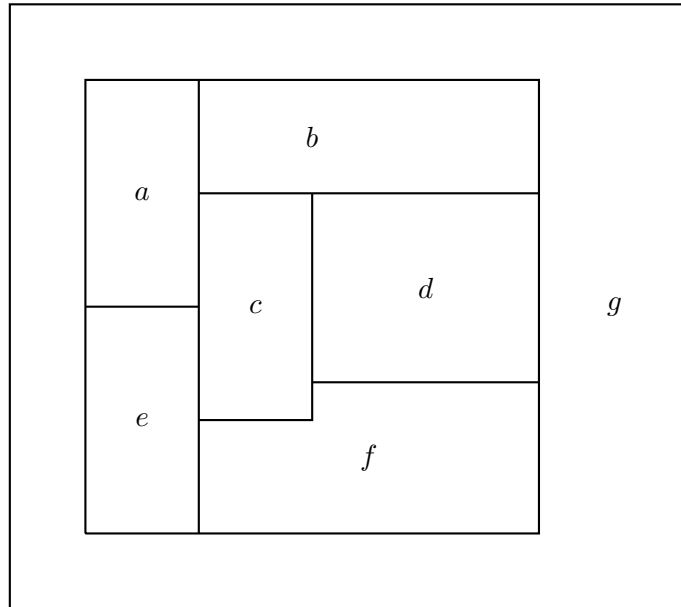
(a) A cartographer wants to colour the map so that no two neighbouring countries have the same colour. How few colours can she get away with?

(b) Show how to reduce the problem to a graph-colouring problem.

5. You have to search for a given number n in a *sorted* list of numbers.

(a) How can you take advantage of knowing that the list is represented as a linked (and sorted) list?

(b) How can you take advantage of knowing the list is represented as an array?

6. In the first lecture we discussed different ways of calculating the greatest common divisor of two positive integers. A mathematician might simply write

$$gcd(x, y) = max\{z \mid \exists u, v : x = uz \wedge y = vz\}$$

and suggest we develop a functional programming language that allows us to write this, leaving it to the language implementation to translate this definition to an efficient algorithm. Do you imagine a time when we may be able to do this? If we restrict our attention to functions like *gcd* which takes a pair of integers and returns an integer, do you think we may some day be able to automatically turn any function definition into a working algorithm?