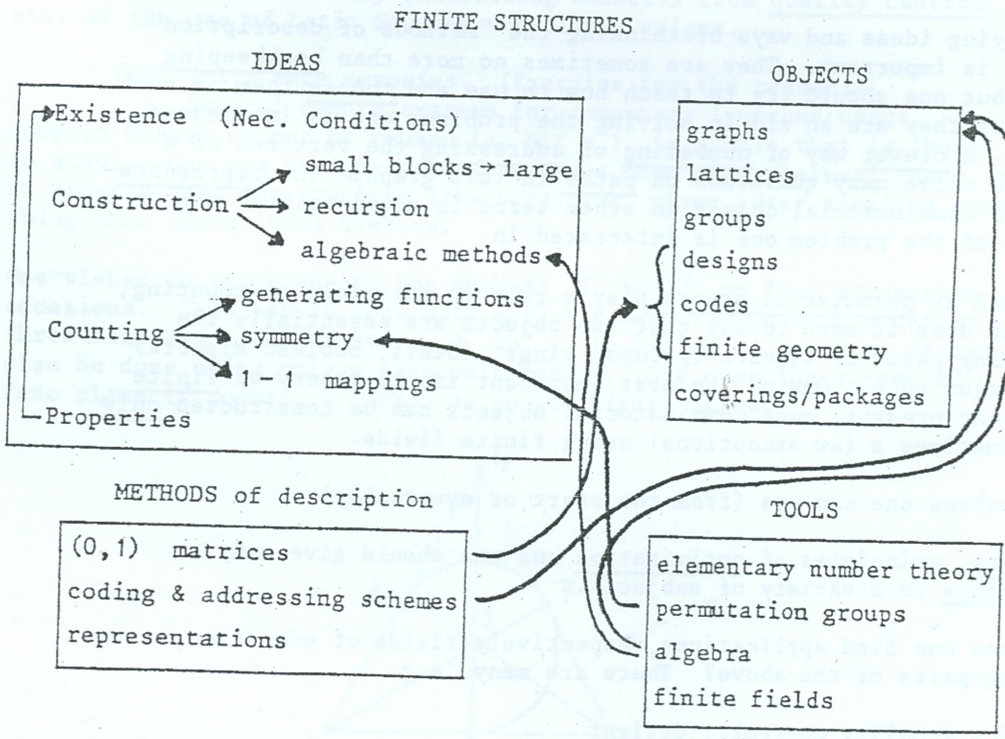
DISCRETE MATHEMATICS: SOME PERSONAL THOUGHTS

J.H. van Lint, Department of Mathematics and Computing Science, Technological University Eindhoven, The Netherlands.

Contrary to what one would think, judging from the recent stream of textbooks on "Discrete Mathematics for Computer Scientists", this subject is not the union of all subjects in mathematics that are not in the calculus course, but necessary for computer science. This note will try to show what discrete mathematics really is.

In the description below we will assume that students have had some calculus, linear algebra (maybe even elementary probability theory) and know what mathematical reasoning is. A course in logic could precede or follow the discrete mathematics; it is not part of it.

We will concentrate on principles, ideas, and the way of thinking which are the essence of discrete mathematics. The following diagram illustrates the comments below.



The main topics in the study of finite structures are existence questions, constructions, counting, and also studying properties of the objects in question.

One should stress the occurrence in many different situations of similar principles, e.g. in construction one has:

a) using several small objects to construct one large one.
b) recursive constructions.
c) using algebraic techniques to construct combinatorial objects.

In counting one uses:

a) generating functions (either ignoring convergence, questions or cleverly using them),
b) symmetry principles (e.g. permutation groups),
c) 1 - 1 mappings of seemingly different objects onto each other.

There are many combinatorial or just "finite" objects to study. Some are mentioned in the diagram. The "tools" are themselves objects of study in discrete mathematics.

In conveying ideas and ways of thinking the "methods of description" are what is important. They are sometimes no more than bookkeeping devices but one should try to teach how to use and choose them in such a way that they are an aid in solving the problems one is interested in. E.g. a clever way of numbering or addressing the vertices of a graph can solve many questions on paths in this graph. The representation of a combinatorial object in other terms is often half of the solution of the problem one is interested in.

Tools such as permutation groups play a rôle in questions of counting, e.g. what does it mean to say that two objects are essentially the same? Many parts of algebra (groups, rings, ideals, boolean algebra) play a major rôle. One of the most important is the theory of finite fields. At present, many combinatorial objects can be constructed only (with sometimes a few exceptions) using finite fields.

In DM courses one can use (from the start or eventually):

algorithms, principles of optimization and one should give many applications to a variety of subjects.

Where does one find applications, respectively fields of education, requiring parts of the above? There are many, e.g.

statistics (quality control): designs
electrical eng. (communication): codes, boolean algebra
computer science: graphs, algorithms, (0,1)-matrices
business administration: graphs, algorithms
social sciences: graphs.

Examples

Many of the subjects taught can be illustrated by examples which most students find fascinating. These examples provide strong motivation. Personally, I prefer giving them after the relevant mathematics has been treated. I mention a few:

(i) Graph-addressing

In some systems a message goes through a telephone network preceded by the address of the destination. At each vertex the message is directed along an edge that brings it closer to the destination. The problem of giving each vertex an address from $<0,1,*>^n$, such that distance of addresses equals distance in the graph (* does not contribute) leads to very interesting problems.

(ii) Switching problems in communication.

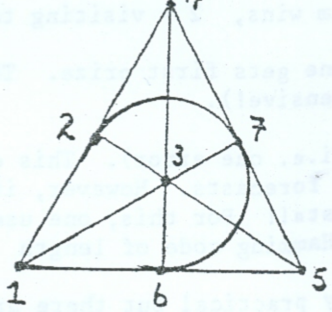
(iii) Hadamard matrices were used to transmit the pictures of Mars made by Mariner '69. It takes less than an hour to show how the quality of the pictures was increased tremendously by the use of an error-correcting code.

(iv) There are many interesting examples from quality control etc. of the use of Latin Squares and Block Designs.

(v) Write once memories. (Exercise for the reader!)

Several memory systems for computers (punched cards, compact disks) cannot be reused (a bit = 1 is a hole that is there to stay). Can one design systems to reuse them nevertheless? Here is an example that the students like very much, given after treating something that maybe looks useless.

One wishes to store one of the numbers 1 to 7 on four successive occasions. This can be done with a 12-bit memory. On each occasion three bits are used (to store 1 to 7 in binary). However, it can also be done using only a seven-bit memory. One uses the so-called Fano plane (7 points, 7 lines) as in the figure below:



(The set {2,6,7} is also a line!)

On the first usage a number, say 5, is stored by punching a hole in position 5. How to proceed on the next three times this memory is used? The reader should try it.

(vi) Conference telephone calls

Electrical engineering students will easily understand the requirements of an electrical network (without resistances) that makes it possible to have a telephone conference with n persons. Each person should be able to hear each of the others equally good (no energy loss, etc. etc.). It takes only a few minutes to translate the requirements into the following:

Is there an n by n matrix C for which all diagonal elements are 0, all the others have the same absolute value (say they are ± 1) and such that any two rows of C have inner product 0? In other words:

$$CC^T = (n-1)\, I .$$

These matrices (called conference matrices (!)) occur in a chapter on designs.

(vii) Search time for data stored in a computer.

We store data having n properties each of which can be one of two kinds (yes = 1, no = 0). The data is stored in batches (or bins, = bin packing). E.g. if all data with first coordinate 0 is in one bin, this bin gets the name (0***...*). One can show that in order to minimize worst-case search time the list of names of the bins is a matrix of 0's, 1's and *'s such that each row (resp. each column) has the same number of *'s, every column has some fixed number of 0's (resp. 1's) and each sequence in $\{0,1\}^n$ belongs to exactly one bin. Clearly a problem from design theory.

(viii) Winning in a football pool

In a football pool one can forecast the outcome of a number of matches (often 13), namely:

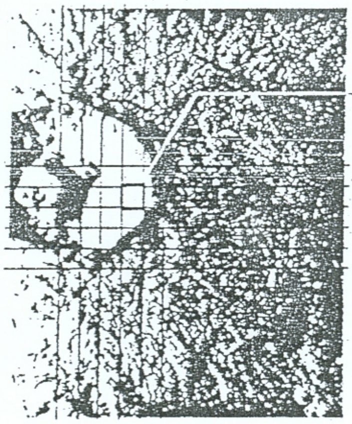
0 = draw, 1 = home team wins, 2 = visiting team wins.

If all 13 forecasts are correct one gets first prize. To guarantee this one needs 3^{13} forecasts (expensive!).

Suppose one aims for second prize (i.e. one error). This can be achieved easily by handing in 3^{12} forecasts. However, it can in effect be done by only 3^{10} forecasts!! For this, one uses coding theory, namely the perfect ternary Hamming code of length 13.

(This idea is still not economically practical but there are practical schemes designed by using these ideas.)

(ix) Pictures of Mars (Mariner '69)



Blackness = 43

43 = 101011

.... 101011 .....

Picture sequence →

A picture is divided into little squares (pixels). For each pixel the degree of blackness is measured (in a scale of 0 - 63, in binary). So, this degree is described by a sequence of six 0's and 1's. The picture results in millions of 0's and 1's to be transmitted to earth.

The transmitted message is corrupted by noise. The effect is that some 0's are interpreted as 1's (and vice versa). As a result the quality of the picture could become extremely bad.

Suppose we are willing to take roughly five times as long to transmit the message. We could repeat each bit five times. This would lead to a substantial improvement but nowhere near to what was achieved in practice. The following solution was used:

A Hadamard matrix of order n is an n by n matrix H with entries ± 1 such that $HH^T = nI$. (These play an important rôle in Combinatorics, turn up quite often and are part of many courses). Construction is easy if n is a power of 2 (induction).

Consider the 64 rows of the two matrices H and $-H$ (H of order 32). To send the number i, transmit row number i. The receiver takes the received message $\underline{x}$ and calculates $\underline{x}\,H^T$. If there are no errors, then the result has 31 coordinates 0 and one coordinate equal to ± 32. Now, suppose there are t errors ($t \leq 7$). The coordinate that should be ± 32 still has absolute value ≥ 18, all others should be 0 but aren't. However, they have absolute value ≤ 14. So the correct value of the darkness can still be established.

In practice, it was extremely unlikely that a sequence of 32 signals contained more than 7 errors. Result: beautiful pictures.