Appendix III

Visual presentation of rules

Many rules, especially legal rules, are very complicated, often involving exceptions, qualifications, provisos and double negatives. In these cases, the reader may be in doubt as to the relationship between the various parts of the rule. In Chapter 1 section 9 we suggested that one way of easing the difficulties and interpreter might face in understanding what a complex rule means is to break the rule down into all of its constituent parts and then present these as a series of options.

Here is an example. Section 67(1) of the Criminal Justice Act 1967 read (note that the section was repealed and replaced by section 240 of the Criminal Justice Act 2003):

> The length of any sentence of imprisonment imposed on an offender by a court shall be treated as reduced by any relevant period, but where he was previously sentenced to a probation order, a community service order, an order for conditional discharge or a suspended sentence in respect of that offence, any such period falling before the order was made or suspended sentence passed shall be disregarded for the purposes of this section.

At first sight, this may seem a little confusing because of its multiple use of subordinate clauses, because it contains two propositions, the second of which has a variety of sub-options, and because it is an unbroken narrative. We can rewrite the section in a schematic form:

1 The length of any sentence of imprisonment imposed on an offender by a court shall be treated as reduced by any relevant period
2 but where he was previously sentenced to:
  (a) a probation order,
  (b) a community service order,
  (c) an order for conditional discharge; or
  (d) a suspended sentence in respect of that offence
3 any such period falling before the
  (a) order was made; or
  (b) suspended sentence was passed
4 shall be disregarded for the purposes of this section.

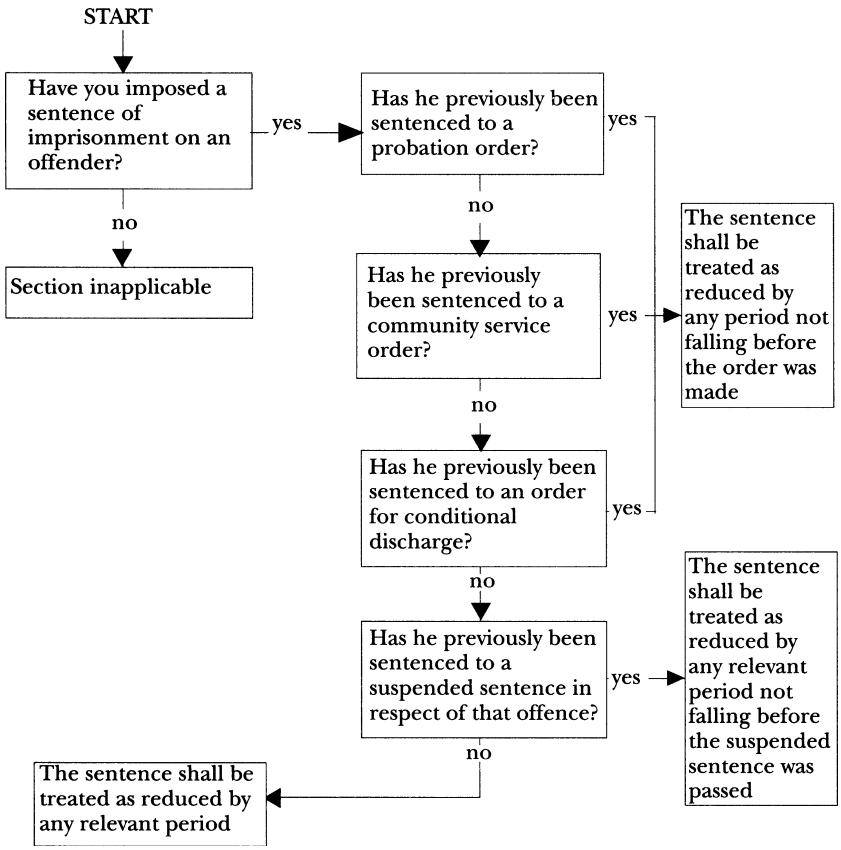
The advantage of this type of presentation over conventional prose style is that it is clearer to the reader. The text should therefore be more easily understood. The layout also provides a simple checklist of the conditions under which the provision operates. It is a technique that can also be used in legislation. See for example section 23 of the Income Tax Act 2007 (giving effect to the Tax Law Rewrite discussed in Chapter 7), which sets out the 'steps' that taxpayers should follow to find their liability to income tax for a tax year.

Another way in which to rewrite the section is in some pictorial form, such as a flow chart, map or other visualization. Flowcharts can be used to identify exactly the steps a person needs to take in order to find out her eligibility for social security or other statutory benefits,[1] liability to tax,[2] and more mundane practical tasks such as taking a bath.[3] A particular visual device that serves this purpose is the algorithm. An algorithm is a precise set of instructions for solving a well-defined problem.[4] It takes the form of a structured series of questions with answers providing instructions for total or partial (when more questions need to be answered) resolution of the problem.

Here is section 67 of the Criminal Justice Act 1967 in algorithmic form.
An algorithm comprises a sequence of questions to which the answer is either 'yes' or 'no', each answer automatically taking the reader to the next question relevant to her case. Either of these answers may take the reader outside the ambit of the rule, or provide a solution to her specific case, in which event there is no need to proceed further. Indeed, in such a case, if the algorithm has been constructed properly it should be impossible to proceed further. This last statement indicates a fundamental attribute of the algorithm: it eliminates choice for the reader of a rule. Provided she can answer the questions, the reader of a rule in algorithmic form should always reach the conclusion appropriate in her case. It follows from this also that the reader may not need to read the whole rule, which might otherwise be a source of confusion for her, for the process of questions and answers should mean that she reads only those parts that are applicable to her case. These attributes of the algorithm follow from the fact that in the algorithm the rule is broken down into a series of questions to which the reader can only answer 'yes' or 'no'. This allows us to state one basic rule for algorithm construction.

(a) To each question there can only be one 'yes' and one 'no'. Each 'yes' or 'no' takes the reader automatically to the next relevant question, but this also means that only one question can follow on each answer. In other words,

[1] I. Goldrein, 'Multipliers and Lump Sum Payments', *New Law Journal*, 48 (1998), 1149, 1237.
[2] See S. Motro, *The Income Tax Map: A Bird's-Eye View of Federal Income Taxation for Law Students* (Thomson, West 7th edn, 2008–9).
[3] See the Centre for Innovation in Mathematics Teaching, at: www.cimt.plymouth.ac.uk/projects/mepres/book8/bk8i1/bk8_1i2.htm
[4] This is adopted from B. Lewis and P. Woolfenden, *Algorithms and Logical Trees* (1969). See also D. M. Wheatley and A. W. Unwin, *Algorithm Writer's Guide* (Longmans, 1972); I. Horabin and B. Lewis (D. Langdon (ed.)), *Algorithms* (Educational Technology Publications, New Jersey, 1978); and T. Cormen, *Introduction to Algorithms* (MIT Press, 2nd edn 2001).



Algorithm designed to instruct a court of the limitation on its power to reduce a sentence of imprisonment where the offender has served a 'relevant period'

following each answer, the reader has no choice as to the next question. This leads to a second basic rule:

(b) There can be only one question following each answer. The rule above is subject to variation where an answer leads to a conclusion (usually called an 'outcome'). In that event, of course, that part of the algorithm is complete, and no more questions need or can be asked.

The function of an algorithm is to present rules in a visually more comprehensible form than conventional prose.[5] An algorithm will not resolve doubts that may arise as to the scope of a rule. In the example used above, if the reader does not know what is meant by 'any sentence of imprisonment', converting the rule into algorithmic form will not help him to resolve his doubt, though in some cases the conversion may help him to identify more closely the *locus* of his

[5] R. Fox, 'Algorithms can take Flow Charts to Next Step for Complex Situations', *Law Teacher,* 3 (1996), 2.

doubt. An algorithm only affects the arrangement of the parts of a rule,[6] and because it cannot resolve doubts arising as to the interpretation of words employed in the rule, or as to the rule's policy, it is defined as a precise set of instructions for resolving a well-defined problem. Converting complex rules into algorithmic form is a useful preliminary to interpretation.

Algorithms can be used both by those who wish to discover the effect of a rule in a particular case and by those teaching and learning about the interpretation of rules. Income tax and VAT returns, house purchase, claims for housing benefit, legal aid, and the like bring the layman into contact with complex rules of law; here the clarity of the algorithm can help him to establish quickly whether the provisions of a particular rule apply to his case, while its structure may eliminate possible error by saving him the trouble of having to understand the whole provision when only a part of it is applicable.[7] Algorithms may also help lawyers to familiarize themselves with the effect of recently published legislative rules; although by virtue of their professional expertise in rule-handling, as a class, they perhaps have less need of assistance.

Algorithms can be a useful educational technique for imparting and acquiring an understanding of the interrelationship of the parts of a rule or of different rules comprising a complex body of law. They are adaptable, so that it is possible to move from simple to more difficult exercises. They can also be used to organize into manageable form large quantities of data or other material,[8] for example, the primary, secondary and other authoritative rules in such areas as company or family law, civil and criminal procedure.[9] They can also be used to show the individual elements of rules not in fixed verbal form, such as, in the example on p. 00, the requisites of the tort of defamation. This simplified statement would of course require supplementation from a standard textbook or practitioner's work.

We do not wish to exaggerate the claims made for algorithms in the context of rule-handling. Their more obvious limitations are that they become

[6] It is therefore a precise tool for identifying syntactic ambiguity; see D. Miers, 'Barking Up the Wrong Tree: Determining the Intention of Parliament', *Statute Law Review*, 13 (1992), 50.

[7] See the algorithm in Chapter 1, section 9. By reason of their structure, algorithms are readily usable in electronic form; see, for example, Ferret *Information Services, Welfare Benefits Advice and Assessment System* (1998), and *Legal Aid PC* (1998).

[8] Compare J. H. Wigmore's technique for analysing masses of evidence: *Science of Judicial Proof* (1937); T. Anderson and W. Twining, *Analysis of Evidence* (1991). A Wigmore chart is used to give a comprehensive picture of the relations between propositions in an inferential argument about a question of fact based on complex evidence. It is more like the picture of the structure of argument in *Allen* (pp. 347 ff.) than an algorithm or a flowchart, in that relations between the nodes 'tend to support' or 'tend to negate'. Wigmore charts are technically a form of 'directed acyclic graphs whose nodes indicate propositions and whose arcs represent probabilistic linkages among nodes': D. Schum, *Evidential Foundations of Probabilistic Reasoning* (1994), pp. 169–70. A chart is like an algorithm in that it is binary and is a pictorial device that can be used to structure material, but the two devices should not be confused.

[9] Examples include S. Wilton, 'Remanding Unruly Juveniles', *New Law Journal*, 139 (1989), 718–19 and 'Structured Decision Making in the Fine Default Court', *New Law Journal*, 138 (1988), 167; and A. Mennie, 'Enforcement of Judgments Flowchart', *New Law Journal*, 138 (1988), 254–5, 'Choice of Law in Contract Flowchart', *New Law Journal,* 148 (1998), 590.

cumbersome when applied to lengthy rules, that they can take a long time to construct, and that their utility wanes as one becomes familiar with particular rules. Algorithms are a tool for promoting skill in the handling of complex rules and, as such, may be dispensed with at times. In addition, the legal examples given in this Appendix to illustrate what use may be made of algorithms and how they are constructed have primarily been isolated sections from statutes. It goes without saying that this is an artificial way of reading statutory rules, which are normally part of a much wider range of provisions, and which need to be read in the light of them. Thus, the algorithms are to varying degrees incomplete, in that they do not explicitly take account of other relevant rules, and an interpreter who is seeking to present a comprehensive statement of a rule would have to account for them, either by including their text in his algorithm, or by referring to them in some other way.

The process of reasoning in algorithms is not unlike a common-sense approach to problem-solving, in that one seeks to eliminate possibilities by adopting some coherent plan; but perhaps the most important aspect of the acquisition of proficiency in reducing a rule from prose to algorithmic form is the intellectual discipline that is involved. Before you can present a rule in this way, you must be in a position to understand the interrelationship of the different parts of the rule, and we conclude this brief discussion with the algorithm (see below) designed to assist algorithm-writers to evaluate their algorithms, and some exercises in constructing an algorithm.

Techniques and methods of visual presentation of legal materials have in recent years moved far beyond algorithms, which still remain useful. A lot of the developments have been inspired by the superb books of Edward R. Tufte. See his *Envisioning Information* (Graphics Press, Connecticut, 1990) and *Visual Explanations* (Graphics Press, Connecticut, 1997). It is beyond the scope of this book to give a systematic account, but the following are links to some useful literature. First, Sharon Hanson's *Legal Method, Skills and Reasoning* (3rd edition, Cavendish, 2010) makes brilliant use of visual presentations to illuminate many of the topics and points developed in *How To Do Things with Rules*. As was noted in the Preface, her book adopts a very similar approach to ours and usefully complements it. A particular application of visual presentations is J. Lowrence, 'Graphical Manipulation of Evidence in Structured Arguments' (2007) 6 *Law, Probability and Risk* 225.

EXERCISES

1 Set out below is section 17 of the Juries Act 1974 as amended which provides for majority verdicts in jury trials. Construct an algorithm designed to tell a trial judge when a majority verdict will be acceptable.

(1) Subject to subsections (3) and (4) below, the verdict of a jury in proceedings in the Crown Court need not be unanimous if—
  a) in a case where there are not less than eleven jurors, ten of them agree on the verdict; and

b) in a case where there are ten jurors, nine of them agree on the verdict.

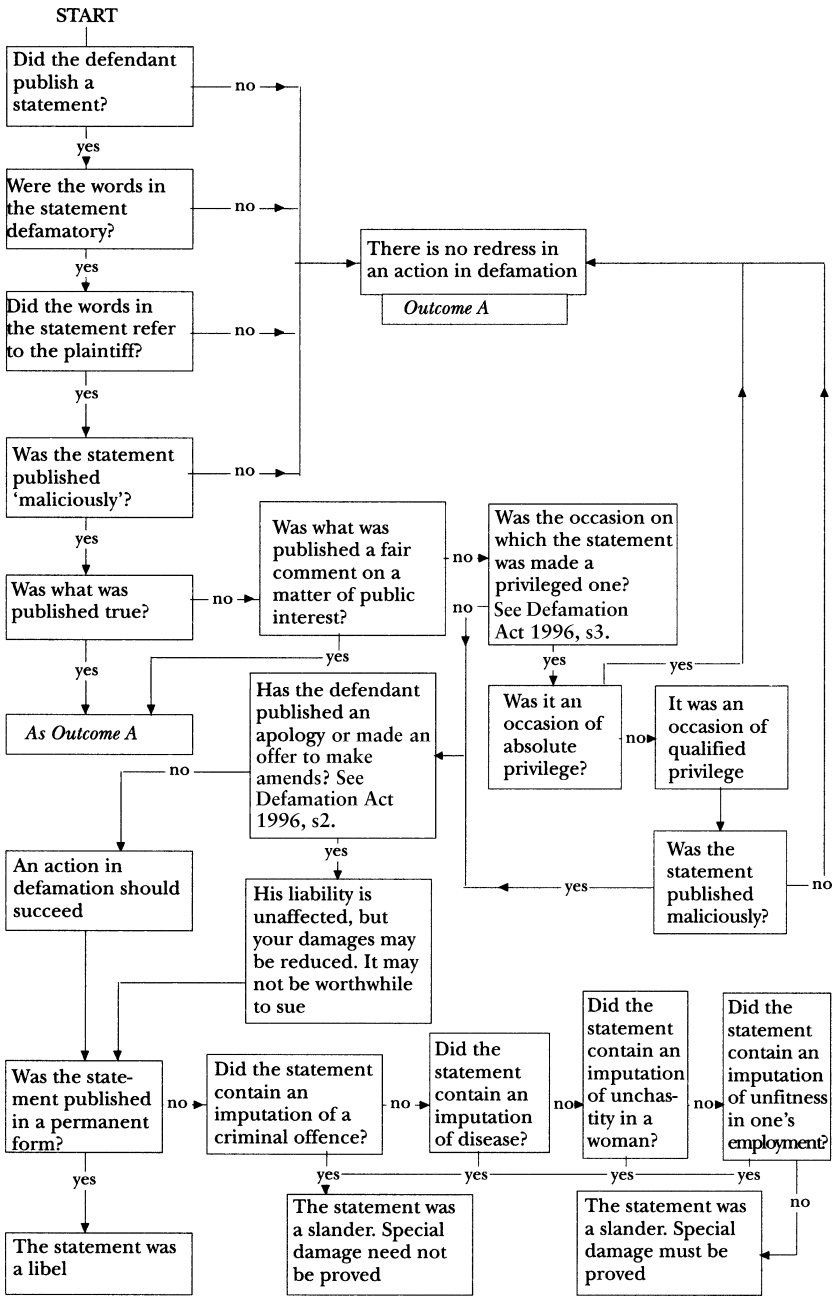
(2) [repealed]

(3) The Crown Court shall not accept a verdict of guilty by virtue of subsection (1) above unless the foreman of the jury has stated in open court the number of jurors who respectively agreed to and dissented from the verdict.

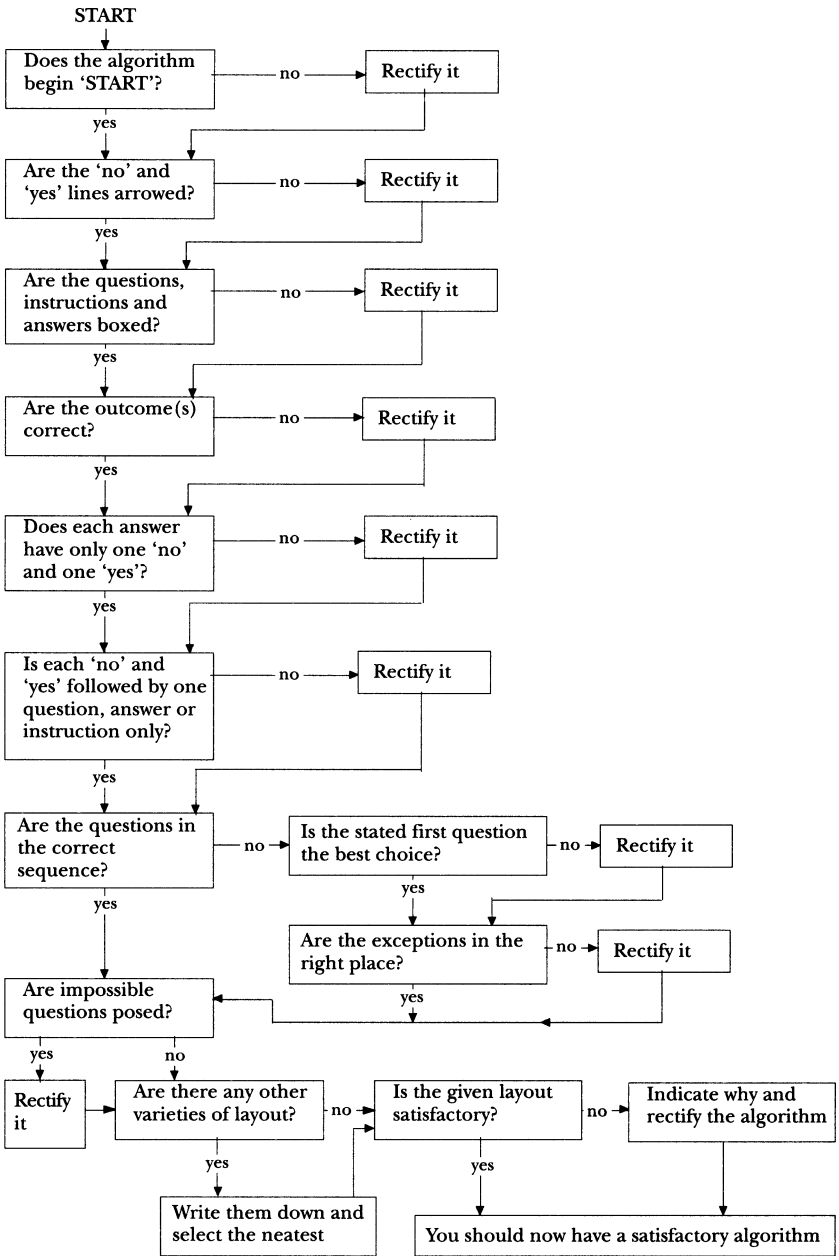
(4) No court shall accept a verdict by virtue of subsection (1) above unless it appears to the court that the jury have had such period of time for deliberation as the court thinks reasonable having regard to the nature and complexity of the case; and the Crown Court shall in any event not accept such a verdict unless it appears to the court that the jury have had at least two hours for deliberation.

2 Construct an algorithm designed to help a law student understand the structure of section 57 of the Offences Against the Person Act 1861 (Chapter 1, section 10.2). Would this have been helpful to you before you read *Allen* and *Taylor* (Chapter 1, sections 10.3 and Appendix 1, section D)? Could an algorithm have helped the judges to interpret section 57 in those two cases?

3 Other exercises will be found in Appendix V.



Algorithm designed to show the main requisites for liability in defamation



Algorithm designed to help algorithm-writers evaluate their algorithms