```
    get bin (binin, X);
    T:= build (1);
    if X # maxint
      then c error in the data c
                  T:= node := (nil, maxint, nil)
                    c maxint signifies error to user program c
      fi;
      T
  end;
```

```
      skip
  end
  keep dump, recreate
  finish
```


References

LARSON, P. (1978). Dynamic hashing, *BIT*, Vol. 18, pp. 184–201.

WOODWARD, P. M. and BOND, S. G. (1972). ALGOL 68-R. Royal Radar Establishment.

Book reviews

Programming Language Standardisation, edited by I. D. Hill and B. L. Meek, 1980; 261 pages, (*Wiley*, £18.50)

This volume succeeds in its claim to fill an important gap in the literature. Most computing people use and are bound by standards throughout their working lives; however, I would assume that few know how standardisation is achieved or who achieves it (or even why?)

The book provides an insight into the faceless business of standardisation, giving the reader a 'walk' through the veritable jungle of committees involved in the process. As a member of the EWICS TC2 committee mentioned in the chapter on BASIC, I certainly found this new world confusing for some considerable time.

The book is a collection of contributions from personalities of long standing in the field of standardisation and divides into two sections. The first provides a series of chapters on the standardisation of a number of well-known languages and functions, they are FORTRAN (Walter Brainerd), COBOL (John Triance), ALGOL 60 (David Hill) PL/I (David Beech and Michael Marcotty), BASIC (Gordon Bull) Pascal (Anthony Addyman) Real-time Languages (Nicholas Neve) Data Base Management Systems (William Olle), Graphics (David Fisher). Operating System Command Languages (Ingemar Dahlstrand), Metalanguages (Roger Scowen) and Flowcharts and Decision Tables (Roger Johnson). Part I is concluded by a chapter by Brian Meek on other potential programming language standards.

This section of the book goes into detail not only on the chronology of standardisation for the various languages etc., but also on some of the technical issues that arose during the process. It is highly readable and provides an interesting documentation of what is often an involved and highly iterative procedure.

Section 2 comprises an edited dialogue between the contributors to Section 1. The discussions are wide ranging and consider topics such as the purpose of standardisation, the form of standards and the relationships between standards. This is a novel way of presenting a set of views and is intended to give a feel for the informal atmosphere in which standards are claimed to be produced (perhaps this is not strictly true in the case of ANSI!). However, the areas ranged across and the contributions made are both valuable and constructive.

There is a need for a book of this nature and the idea of drawing together interested and active contributors in this way is probably the best method of producing an authoritative and informative text. I would feel that the volume is a must for any computing library although its cost (£18.50) may place it outside the range of the individual purchaser (I look forward to the paperback version?)

R. W. NEWTON (Middlesbrough)

Problem Solving and Structured Programming in PASCAL, by E. B. Koffman, 1981. (*Addison-Wesley*, £6·95)

This book is a good tutorial both for learning and teaching programming.

Standard Pascal features are extensively described and fully discussed through a wide range of examples. The proposed exercises also constitute a valuable contribution to the learning process.

F. PANZIERI (Newcastle-Upon Tyne)

Data Base: Structured Techniques for Design, Performance and Management, by S. Atre, 1980; 442 pages. (*Wiley*, £15·00)

With 442 pages, it is difficult for a reviewer to say that the book does not cover enough depth. Similarly, with over 200 figures, it is difficult to say that the exposition is not always very clear. However, that is precisely the impression with which one is left. Part of the problem is that the 200 plus figures occupy between a quarter to a third of the space. Too much of a good thing. Second, in the various kinds of data diagrams, there is no uniform convention. Sometimes rectangles represent types of entities, (e.g. all bank customers), at other times the rectangles represent instances of entities, (e.g. savings account 8943211). Yet again, in some chapters circles represent types, (e.g. Surgeon) or instances, (e.g. Patient—John White). The same applies to arrow-heads representing relationships. Sometimes ——→ represents 1 to many relationships, whereas at other times ←—→→ represents 1 to many. In the same way, no diagrammatic distinction is made between conceptual data diagrams and implementation data diagrams.

A second reason for the criticism of lack of depth and clarity is that in attempting to be practical, the author covers too much ground. Not only are hierarchical, network and relational techniques covered, occasionally inverted structures such as ADABAS and old-fashioned file handling—sequential, indexed, hashed, etc. are also touched upon. The net result is that the material is too complex for the student and too shallow for the practitioner.

The reviewer is left with a suspicion that the book is a quick repackage of the author's course notes. If this is the case, much of the repetition and overlap between chapters and the overabundance of diagrams would be explained. The author probably overcomes the deficiency through the lectures, thereby providing adequate depth. Having said all this, as a data base person, the reviewer could not find anything wrong, inaccurate or misleading in all of 400 plus pages! The chapters are logically organised. In many ways, the various checklists in some chapters such as DB Administration and Data Dictionary are useful. Similarly the 100 page appendices which takes the reader through two 'case studies' are useful to a database design methodology.

Overall: worth browsing through—but not an acquisition for one's personal library.

C. C. CHANG (London)

Pascal—The Language and its Implementation, edited by D. W. Barron, 1981; 301 pages. (*Wiley*, £12·50)

This book is mainly a collection of papers from a symposium of the same name, which was held in March 1977! In addition to the symposium papers, the ETH Zurich publications on Pascal-P, Pascal-S and the CDC implementation are included, together with a paper from Software Practice and Experience.

There is little evidence of any attempt to update the symposium papers to reflect the progress made since March 1977. Given the delay in publication, it is difficult to see why anyone should want to buy this book. Pascal devotees will find the preface interesting to read. It is inCREDible.

A. M. ADDYMAN (Manchester)