

KAREL THE ROBOT
by Richard Pattis

John Wiley Sons, 1981, 106 pgs., (paperback) \$9.50
Reviewed by Harry E. Pence*

The fundamental concept of this book will be immediately obvious to anyone who has previously had the pleasure of using LOGO. Like the familiar LOGO turtle, Karel. the Robot is simply another name for the cursor on the computer screen. His movements can be controlled

by a relatively simple programming language. The subtitle of the book, 'A Gentle Introduction to the Art of Programming with Pascal' indicates that Pattis has written his book not merely to provide pleasure for the reader, which it does, but also to provide an introduction to the fundamental concepts of Pascal.

The ground rules of Karel's universe are simple. The screen is divided into a ten by ten grid, described as streets and avenues. Movement across this grid is restricted by the presence of walls which are placed at different positions in each problem. Karel can move forward one grid space at a time, turn to his left, pick up or deposit markers ("beepers") in positions next to the cursor, and finally he can be turned off when the task is completed. If he is given an order which is impossible to execute, he turns himself off.

In each problem, the purpose is to move Karel across the screen, either picking up or depositing beepers in a predetermined fashion. Although the initial situations are relatively straight-forward, the problems become increasingly complex and are positively challenging by the end of the book. Students will find the approach to be non-threatening and conducive to learning. Numerical data, input-output methods, and data structures are de-emphasized, so that the student can develop a good foundation in the essentials of programming and problem solving.

Starting from the the primitive control structures described above, the

instruction set becomes increasingly sophisticated, allowing for the development of block structures which are essentially equivalent to procedures in Pascal, conditional statements, such as IF/THEN/ELSE, and finally pretest loops, using the WHILE-DO command. Each of these new commands is integrated into a modular structure which closely resembles that used in Pascal.

The author has not only written a very readable introduction to structured programming, but he has also combined this with an excellent discussion of problem solving strategies. The method of stepwise refinement is introduced early and is employed consistently throughout the book. Pattis uses boxes of block diagrams to help the student to visualize the program structures, and he emphasizes that programs must be clear and understandable. The book doesn't teach Pascal, but a student who has completed this book will certainly find Pascal to be familiar and easy to learn.

Both mainframe and microprocessor versions of Earel are available, but personal experience suggests that this support is not essential. Running the programs on a computer does help to catch subtle errors, but the author is quite right when he suggests that if time is limited, it is equally effective to do the programs by hand. The preface indicates that this entire book is covered in four class days at Stanford. This seems a little rapid for those who have never programmed before, but it accurately suggests that the book is clearly written and easy to understand.

Pattis has written a book which is not only fun to read but also teaches many of the basic ideas of computer programming. It is a good introduction to structured programming for the beginner and can also serve as a transition to Pascal for someone who is already familiar with a non-structured language, such as BASIC. It is a pleasure to recommend this delightful little book.

*Department of Chemistry

SUSY-Oneonta, Oneonta, NY 13820