Our exploration of computer systems starts by studying the computer itself, comprising a processor and a memory subsystem. At

the core, we require ways to represent basic data types, such as

approximations to integer and real arithmetic. From there, we can consider how machine-level instructions manipulate data and how a compiler translates C programs into these instructions. Next, we study several

methods of implementing a processor to gain a better understanding of

how hardware resources are used to execute instructions. Once we understand compilers and machine-level code, we can examine how to maximize program performance by writing C programs that, when compiled,

achieve the maximum possible performance. We conclude with the design of the memory subsystem, one of the most complex components of

a modern computer system.

This part of the book will give you a deep understanding of how

application programs are represented and executed. You will gain skills

that help you write programs that are secure, reliable, and make the best

use of the computing resources