

A Dictionary of
Computing

FIFTH EDITION

OXFORD
UNIVERSITY PRESS

OXFORD

UNIVERSITY PRESS

Great Clarendon Street, Oxford OX2 6DP

Oxford University Press is a department of the University of Oxford.
It furthers the University's objective of excellence in research, scholarship,
and education by publishing worldwide in

Oxford New York

Auckland Bangkok Buenos Aires Cape Town Chennai
Dar es Salaam Delhi Hong Kong Istanbul Karachi Kolkata
Kuala Lumpur Madrid Melbourne Mexico City Mumbai Nairobi
São Paulo Shanghai Singapore Taipei Tokyo Toronto

Oxford is a registered trade mark of Oxford University Press
in the UK and in certain other countries

© Market House Books Ltd. 1983, 1986, 1990, 1996, 2004

The moral rights of the author have been asserted

Database right Oxford University Press (maker)

First published 1983

Second edition 1986

Third edition 1990

Fourth edition 1996

Fifth edition 2004

All rights reserved. No part of this publication may be reproduced,
stored in a retrieval system, or transmitted in any form or by any means,
without the prior permission in writing of Oxford University Press,
or as expressly permitted by law, or under terms agreed with the appropriate
reprographics rights organization. Enquiries concerning reproduction
outside the scope of the above should be sent to the Rights Department,
Oxford University Press, at the address above

You must not circulate this book in any other binding or cover
and you must impose this same condition on any acquirer

British Library Cataloguing in Publication Data

Data available

Library of Congress Cataloging in Publication Data

Data available

ISBN 0-19-860877-2

1

Typeset in Swift by Market House Books Ltd.

Printed in Great Britain by Clays Ltd, St Ives plc

/.....*/* in C.

Some languages, including Ada, prefer “end-of-line” comments, which are introduced by a characteristic symbol and are automatically terminated at the end of a line. Older languages such as Basic and Fortran restrict comments to be whole lines and do not allow them to be appended to a line of code.

common application environment

(CAE) The set of services available to an application program on a Macintosh computer, defined in such a way that these services can be implemented on a different platform.

COMMON area In Fortran, an area of storage accessible from more than one program unit (subroutine). Data is local to the subroutine in which it is defined unless it is declared to be in a COMMON area. There is one anonymous COMMON area, called “blank COMMON”, and any number of other areas, which are named (“labeled COMMON”).

common carrier In the US, a private business or corporation that offers to the public general communication services such as telephone or intercomputer communications. Common carriers are regulated by the Federal Communications Commission (FCC), and all services offered must charge according to tariff schedules filed with and approved by the FCC.

Common Criteria An internationally agreed set of criteria for *security evaluation, ratified as ISO standard 15408.

common gateway interface (CGI) A protocol for exchanging data between a Web server and an application. For example, if a user fills in a form on a Web page and submits it to the server, the server passes the data to an application program, which analyzes it. The program may also create a reply to the user, which initially goes to the server. The CGI is a standard way of passing this information from server to program and back again. It is part of HTTP. CGI programs are often written in *Perl.

common instance See UNIFICATION.

Common LISP A version of *LISP that integrates the facilities of FranzLisp and

MACLisp, adopted as an informal standard by the major users and suppliers of LISP systems.

common-sense reasoning Most reasoning in *artificial intelligence concerns higher-level functions, such as game playing, language processing, and symbolic problem solving. Common-sense reasoning is concerned with the understanding and manipulation of information about the everyday world of objects and their interactions. Deciding that a pile of objects is unstable or that a vehicle will not get through a red light in time are examples of common-sense reasoning. Unfortunately this is surprisingly difficult to automate as the problem domain is so ill defined and open-ended; for example, the *closed-world assumption is invalid. One approach is to provide large amounts of domain knowledge (see CYC PROJECT), while others concentrate on reasoning about materials, physics, space, and time (see NAIVE PHYSICS, IMPRECISION). The topic has also stimulated research into philosophical and logical issues in a search for formal structures.

common user access See CUA.

comms (or coms, com) Short for communications, as in datacomms, telecomms, comms equipment, and comms link.

communicating sequential processes See CSP.

communication channel See SHANNON'S MODEL.

communication interface A computer *interface designed to allow connection to other digital equipment. This may be by way of a modem and telephone lines or through a high-performance computer network as in workstation environments. See also COMMUNICATION PORT.

communication network See COMMUNICATION SYSTEM, NETWORK.

communication port (com port) Any external socket on a computer that can be connected to a communication line and used to exchange information with other computers or electronic systems. Communication ports can be *serial ports such as *RS232C, *parallel ports such as the *Centronics or *SCSI interfaces, or ports for

C

stack, and horizontally between corresponding layers in the two remote stacks.

The primary objective of the seven-layer model is to provide a flexible means of describing the behavior of communications systems, capable of dealing with all existing and future technologies, rather than to provide a specific set of protocols and interfaces. The process of reaching international agreements that meet the conflicting requirements of different groups of end-users, accommodate the interests of competing commercial suppliers, and define technically sound products is slow, often taking several years to reach a final set of recommendations. When agreement on proposals has been achieved, the development of commercially viable products conforming to the proposals is again time-consuming. It is then necessary to test the products for conformance, to demonstrate successful interworking between products from different suppliers, and to resolve discrepancies where systems that each separately appear to conform with the proposals do not interwork correctly.

Inevitably during this time, individual users or suppliers will have made their own systems, and in this sense the international standards will always lag behind the ad hoc or proprietary systems. Despite this, the model has itself been used as the basis for networks themselves (X25 for packet networks, ISO 8802.3 for CSMA/CD, ISO 8802.7 for slotted rings, ISO 8802.5 for token rings, ISO 8802.4 for token bus, and X75 for internetwork communication), and for some applications such as electronic mail (X400), directory services (X500), and manufacturing automation (*MAP and *TOP).

S-gate Another name for ternary threshold gate.

SGML Abbrev. for standard generalized mark-up language. An international standard *metalanguage (ISO 8859) used for defining the syntax of textual *mark-up languages. This enables both sender and receiver of the text to identify its structure (e.g. title, author, header, paragraph, etc.). See also DTD, HTML.

shading model See LIGHTING MODEL.

shadow buffer A shadow-testing accel-

eration scheme where each point light source is surrounded by a direction cube, which contains a list of the objects visible through each cell. The direction of an illumination ray is first looked up in the light buffer associated with that light source as a means of fast shadow testing.

shadow-mask cathode-ray tube A type of *cathode-ray tube that has a perforated mask between the electron guns and the phosphor-coated surface to ensure that each electron gun can only hit phosphor spots of the appropriate color.

shaft of light The light component in a liquid that is generated by the diffraction of direct sunlight by the surface of the liquid and may cause light to be concentrated.

Shannon diagram of a *communication system. A diagram illustrating *Shannon's model of such a system, embodying the source, encoder, channel, noise source, decoder, and destination of information.

Shannon-Fano coding (Fano coding)

See SOURCE CODING.

Shannon-Hartley law See CHANNEL CODING THEOREM.

Shannon's model of a *communication system. A widely accepted model, set down by Claude Elwood Shannon in 1948, that has an *information source* sending a *message* to an *information destination* via a medium or mechanism called the *channel*. According to Shannon, "the fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point."

In general, the channel will distort the message and add *noise to it. In order to avoid the distortion, and to reduce the effect of the noise to any desired degree, an *encoder is placed between the source and the channel, and a *decoder is placed between the channel and the destination. Now, the source sends the *transmitted message*, which is encoded as the *transmitted signal*; this is sent through the channel. It emerges as the *received signal*, which is decoded to give the *received message*; this arrives at the destination.

The channel is considered to have a *noise source* that inputs "information" in addi-

S

tion to that in the transmitted signal. The aim of the encoder and decoder is to make the received message resemble, as closely as required, the transmitted message, in spite of the "information" from the noise source.

See also SOURCE CODING THEOREM, CHANNEL CODING THEOREM.

Shannon's theorems See SOURCE CODING THEOREM, CHANNEL CODING THEOREM.

shape blending See BLENDING.

shared logic system A term sometimes used to refer to a system where several terminals share a CPU simultaneously.

shared memory The use of the same portion of memory by two distinct *processes, or the memory so shared. Shared memory is used for interprocess communication and for purposes, such as common subroutines, that lead to compactness of memory. See also MULTIPORT MEMORY, CONCURRENCY.

shareware Software freely distributed with the expectation that anyone wishing to use it on a long-term basis will send a fee to the author. The method of payment and magnitude of the license fee normally forms part of the accompanying documentation. The incentive to register the software may be the promise of future versions, documentation, or extensions to the software.

SHE Abbrev. for safety, health, and ergonomics.

S

sheet feeder A device that may be attached to a printer allowing individual sheets of paper to be fed into the printer without operator action. This device is often available as an add-on feature with impact printers, but is usually incorporated into the mechanism with modern nonimpact printers.

shell A program that provides the *user interface of an *operating system and is often considered to be part of it. The main inner part of the operating system, the *kernel, is thus enclosed by the shell, as in a nut. Some operating systems have a choice of shells.

Shell's method (diminishing increment sort) A sorting algorithm proposed by Donald Shell in 1959 and published as

shellsort. It is a variant of *straight insertion sort that allows records to take long leaps rather than move one position at a time. It does this by sorting each group $G^{(i)}$ of records a distance h_i apart within the file. (The $G^{(i)}$ are *disjoint and together contain all the information in the file.) This is repeated for a decreasing sequence of values h_i , and consequently increasing number of groups $G^{(i)}$, finally ending with $h_i = 1$.

shellsort See SHELL'S METHOD.

shielded twisted pair (STP) See TWISTED PAIR.

shift 1. To change the interpretation of characters. The term is commonly met on normal typewriters as a change from lower to upper case.

2. Any complete set of characters obtainable without shifting. Hence *change shift* is a synonym for shift (def. 1).

3. The movement of a bit pattern in a bit string. A *left shift* of m ($< n$) bits will move the bit pattern in a string

$b_1 b_2 \dots b_n$

leftward, giving

$b_{m+1} \dots b_n ? \dots ?$

Similarly, a *right shift* of m bits converts

$b_1 b_2 \dots b_n$

to $? \dots ? b_1 b_2 \dots b_{n-m}$

The bits that are introduced (shown here as question marks) and the use of the bits that are shifted off the end of the string depend on the kind of shift: *arithmetic*, *logical*, or *circular*. In an arithmetic shift the bit strings are regarded as representations of binary integers; if the leading m bits that are lost are all zero, a left shift of m bits is equivalent to multiplication by 2^m and a right shift can be interpreted as integer division by 2^m . In logical shifts the bits introduced are all zero. In circular shifts the bits shifted off at one end are introduced at the other.

shift character Any character used in a stream of characters to change *shift, i.e. to change the interpretation of the characters. Compare ESCAPE CHARACTER.

shift counter A *synchronous counter that consists of clocked *flip-flops arranged as a *shift register. Data is propa-

sigma algebra (Σ -algebra) *See* SIGNATURE.

sigma language (Σ -language) *See* FORMAL LANGUAGE.

sigma tree (sigma term, Σ -tree, Σ -term) *See* TREE LANGUAGE, TERM.

sigma word (Σ -word) *See* WORD.

sign A means used to distinguish between positive and negative numbers. In a computer there are a number of ways of representing the sign of a number, each of which makes use of a single bit called the *sign bit*. The most obvious way of representing positive and negative integers in computer words is by means of the *signed-magnitude* (or *sign-and-magnitude*) representation. Here, the leftmost bit in a word is used to denote the sign (0 for + and 1 for -) and the remaining bits in the word are used to represent the magnitude of the integer. It is more usual, however, for a computer to use the two's complement representation of integers. *See* RADIX COMPLEMENT, COMPLEMENT NUMBER SYSTEM.

signal 1. A form of data that is usually envisaged as a sequence of values of a scalar quantity – the *amplitude* – recorded (i.e. measured, tabulated, or plotted) against time. The amplitude is most often, but by no means always, an electric potential. For example, if the signal is a sine wave with a mean value of zero (symmetrical about zero volts), then the amplitude is the value of the peak voltage. *See also* DISCRETE AND CONTINUOUS SYSTEMS, SPACE DOMAIN.

2. Any path transmitting some data or function. *See also* CONTROL SIGNAL.

3. To convey data or control information from one point to another.

signal conditioning *Filtering a continuous signal.

signal operation *See* SEMAPHORE.

signal processing The processing of *signals by means of hardwired or programmable devices, the signals being regarded as continuous or discrete and being approximated by analog or digital devices accordingly (*see* DISCRETE AND CONTINUOUS SYSTEMS). *Filtering and *image

processing are examples of signal processing. *See also* DIGITAL SIGNAL PROCESSING.

signal-to-noise ratio The ratio of the *signal power to the *noise power in a physical transmission channel; it is often measured in decibels (dB). The definition is best applied to statistically well-behaved noise such as *white *Gaussian noise. A more useful definition, frequently employed when small signals are needed, is *signal-plus-noise to noise ratio*. This has the benefit of being easy to measure. With large signals and low-noise circuits the two definitions are virtually identical. *See also* CHANNEL CODING THEOREM.

signature 1. A collection of symbols intended to be associated with *sets and with *functions on and elements from the sets. Signatures provide names for the carrier sets, operations, and constants of *algebras. They are central in the precise treatment of the syntax of many computer science concepts, including (a) *abstract data types, (b) *algebraic specifications, and (c) classes, *modules, and *objects. Typically, headers of modules are signatures. The specification of an Ada *package is in effect a signature.

In its simplest form a signature is a set Σ of symbols with, for each $\sigma \in \Sigma$, a natural number $\rho(\sigma)$ called the *arity of σ . A Σ -algebra consists of a set A (called the *carrier* of the algebra) together with, for each $\sigma \in \Sigma$, an n -argument function over A , where $n = \rho(\sigma)$. As an example, suppose

$$\Sigma = \{\text{'zero'}, \text{'one'}, \text{'plus'}, \text{'times'}\}$$

$$\text{with } \rho(\text{'zero'}) = \rho(\text{'one'}) = 0$$

$$\text{and } \rho(\text{'plus'}) = \rho(\text{'times'}) = 2$$

Then one Σ -algebra results from taking the set of all integers as carrier, and associating the number 0 with 'zero', 1 with 'one', addition with 'plus', and multiplication with 'times'. As indicated by ρ , addition and multiplication are 2-argument functions while zero and one, being constants, expect no arguments and their arity is 0.

It is important to realize that the above example describes only one possible Σ -algebra. For example, the carrier could be the real numbers; or, perversely multiplication could be associated with 'plus' and addition with 'times'; equally sets