

Conditional Control Transfer

Encyclopedia Article

Conditional Control Transfer

The following sections of this BookRags Literature Study Guide is offprint from Gale's For Students Series: Presenting Analysis, Context, and Criticism on Commonly Studied Works: Introduction, Author Biography, Plot Summary, Characters, Themes, Style, Historical Context, Critical Overview, Criticism and Critical Essays, Media Adaptations, Topics for Further Study, Compare & Contrast, What Do I Read Next?, For Further Study, and Sources.

(c)1998-2002; (c)2002 by Gale. Gale is an imprint of The Gale Group, Inc., a division of Thomson Learning, Inc. Gale and Design and Thomson Learning are trademarks used herein under license.

The following sections, if they exist, are offprint from Beacham's Encyclopedia of Popular Fiction: "Social Concerns", "Thematic Overview", "Techniques", "Literary Precedents", "Key Questions", "Related Titles", "Adaptations", "Related Web Sites". (c)1994-2005, by Walton Beacham.

The following sections, if they exist, are offprint from Beacham's Guide to Literature for Young Adults: "About the Author", "Overview", "Setting", "Literary Qualities", "Social Sensitivity", "Topics for Discussion", "Ideas for Reports and Papers". (c)1994-2005, by Walton Beacham.

All other sections in this Literature Study Guide are owned and copyrighted by BookRags, Inc.

Contents

Conditional Control Transfer Encyclopedia Article.....	1
Contents.....	2
Conditional Control Transfer.....	3



Conditional Control Transfer

Conditional control transfer was proposed by **John von Neumann** in the mid 1940s. His concept held that a program could be written in a series of subroutines--small blocks of **code** that could be read in any desired order; instead of having one large routine or a single series of chronologically ordered steps.

Since this proposal, conditional control transfer has become a reality. Computer code can branch based on logical statements such as IF...THEN, and can be looped (or repeated) with an FOR statement. Such statements have created libraries of code, also called subroutines, that can be used and re-used.

Conditional control transfer permits code (the subroutines) to be used in various parts of a program, and eliminates the need for a programmer to redundantly write in the same code in different routines. Moreover, the outcome of the use of the **subroutine** in different areas of a program may be different. The subroutine can be executed differently. This is because the machine instruction mechanism of conditional control transfer allows the program sequence to be stopped and started again at any point. Because the instruction programs are stored together with the **data** in the same **memory**, the instructions can be arithmetically changed in the same manner as data can be changed.

As a result of conditional control transfer, computational work has become much faster, flexible and more efficient since von Neuman's time. Regularly used subroutines now do not have to be reprogrammed for each new program in which they are used. Rather, the subroutines can be maintained in the libraries, to be read into memory only when they are required. Thus, a given program might be almost completely assembled using **library** subroutines.