Procedural Abstraction Encyclopedia Article

Procedural Abstraction

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.

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".

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".

Contents

Procedural Abstraction

Procedural **abstraction** is the process of converting a specific procedure into a general procedure by ignoring certain details. This is used during the **design** process to allow the programmer to focus on the structure of the program instead of the detail of the individual functions.

A procedural abstraction specifies everything the users require from the procedure but nothing more. All other details are left up to the programmer to determine during the implementation. Procedural abstractions are sometimes referred to as "black boxes" since they describe what a procedure does without describing how it does it.

Defining a procedural abstraction requires two steps. The first step names the **input and output** parameters as well as their **types**. The second step defines the conditions (called the requires clause), any side effects (the modifies clause) and what the abstraction achieves (the effects clause).

When a procedural abstraction is defined completely, it has the properties of locality and modifiability. Locality means that the details of the implementation are local to the individual procedure and only need to be known by the programmer dealing with the implementation. Modifiability means that replacing the implementation of the procedure does not affect the rest of the program.