

# Depuração

Origem: Wikipédia, a enciclopédia livre.

**Depuração** (em inglês: *debugging*, *debug*) é o processo de encontrar e reduzir defeitos num aplicativo de software ou mesmo em hardware. Erros de software incluem aqueles que previnem o programa de ser executado e aqueles que produzem um resultado inesperado.

## Índice

- Origem
- Ferramentas
- Processo
- Referências
- Bibliografia
- Ver também

## Origem

Existe alguma controvérsia em relação a quem usou pela primeira vez o termo *bug* -- mas é possível que seu uso tenha se dado anteriormente ao famoso uso feito por Grace Hopper, notável desenvolvedora da Marinha dos Estados Unidos.

Quanto ao termo *debug*, aceita-se em geral que foi usado pela primeira vez por Glenford J Myers no seu livro, *Software Reliability: Principles and Practices*, em 1976.

## Ferramentas

De forma geral, a depuração é uma tarefa difícil e trabalhosa, e a dificuldade varia de acordo com o ambiente de desenvolvimento, o que inclui a linguagem de programação e as ferramentas disponíveis, como depuradores. Depuradores são ferramentas que permitem ao programador monitorar a execução de um programa, pará-lo e reiniciá-lo, ativar pontos de parada, alterar áreas de memória e, em alguns casos, voltar no tempo.

De forma geral, linguagens de alto nível tornam a depuração mais fácil, pois fornecem mais ferramentas para identificar erros, como o tratamento de exceções. Em linguagens de baixo nível, erros de código podem causar problemas difíceis de serem identificados, como corrupção de memória. Nesse caso, depuradores de memória podem ser necessários.

Para certos tipos de problema existem ferramentas de análise do código fonte, que buscam por erros específicos no código, o que depende da linguagem de programação em uso. Enquanto um compilador se preocupa com a sintaxe do código fonte, tais ferramentas de análise focam a semântica. Um problema geralmente identificável através do código fonte é o uso de uma variável antes da primeira atribuição.

Para a depuração de hardware, assim como software de baixo nível (como BIOS, firmware e drivers de dispositivo) instrumentos como osciloscópios, analisadores lógicos e emuladores são frequentemente usados.

## Processo

A depuração começa com a tentativa de reprodução do problema, o que pode não ser uma tarefa simples, como em computação paralela. Após a reprodução, o problema deve ser reduzido até sua essência, para facilitar a depuração. É um processo iterativo em que para cada redução, uma nova execução é feita para assegurar a reprodução do problema. Como analogia, pode-se considerar

esse processo de redução como uma forma de divisão e conquista. Para automatizar a redução da entrada, métodos de depuração delta podem ser usados.<sup>[1]</sup> A redução pode ser feita com a simplificação das entradas do programa. Por exemplo, um programa que falha na leitura de milhares registros pode ter o problema associado a somente um deles, somente um registro possui valores que fazem o problema se manifestar; pode-se reduzir a quantidade de registros lidos até que reste somente um, o causador do problema. No caso de erro de software, a redução é chamada de isolamento de código. Após a identificação dum problema em certo código, pode-se percorrer o código fonte para isolar a origem do problema, a partir de módulos, passando por subrotinas e chegando na linha exata.

Após a reprodução do problema estar suficientemente reduzida (isolada), deve-se identificar o problema, num processo de dedução e teste da origem do problema.<sup>[2]</sup> Para isso, o programador pode usar um depurador para examinar e alterar os estados do programa (valores de variáveis, pilha de chamada). Alternativamente, pode-se realizar um *trace*, um método de registro da execução de um programa. Em um caso simples, um *trace* registra os valores de variáveis em certos pontos de execução do programa. Entretanto, assim como o processo de redução de entrada antes da depuração, deve-se realizar um processo de redução de variáveis registradas antes do *trace*, a fim de reduzir a quantidade de informação gerada, tornando o processo mais objetivo. A depuração remota é o processo de depuração de um programa sendo executado em um sistema diferente do depurador, numa conexão feita geralmente através duma rede.

## Referências

- Zeller, p. 123.
- Parr, 2004

## Bibliografia

- David J. Agans (2002). *Debugging*. The Nine Indispensable Rules for Finding Even the Most Elusive Software and Hardware Problems. [S.l.]: AMACOM. ISBN 0-8144-7168-4
- Bill Blunden (2003). *Software Exorcism*. A Handbook for Debugging and Optimizing Legacy Code. [S.l.]: APress. ISBN 1-59059-234-4
- Frederick Phillips Brooks (1995). *The Mythical Man-Month*. Essays on Software Engineering. [S.l.]: Pearson Addison Wesley. ISBN 0-201-00650-2
- Ann R. Ford, Toby J. Teorey (2002). *Practical Debugging in C++*. [S.l.]: Prentice Hall. ISBN 0-13-065394-2
- Robert Metzger (2003). *Debugging by Thinking*. A Multidisciplinary Approach. [S.l.]: Digital Press. ISBN 1-55558-307-5
- Glenford J Myers (2004). *Software Reliability*. Principles and Practices 2 ed. [S.l.]: John Wiley & Sons inc. ISBN 0-471-62765-8
- Glenford J Myers (2004). *The Art of Software Testing*. [S.l.]: John Wiley & Sons inc. ISBN 0-471-04328-1
- John Robbins (2000). *Debugging Applications*. [S.l.]: Microsoft Press. ISBN 0-7356-0886-5
- Matthew A. Telles, Yuan Hsieh, Matt Telles (2001). *The Science of Debugging*. [S.l.]: The Coriolis Group. ISBN 1-57610-917-8
- Andreas Zeller (2005). *Why Programs Fail*. A Guide to Systematic Debugging. [S.l.]: Morgan Kaufmann. ISBN 1-55860-866-4
- Terence Parr (7 de dezembro de 2004). «Learn the essentials of debugging» (http://www-128.ibm.com/developerworks/web/library/wa-debug.html?ca=dgr-lnxw03Dbug) (em inglês). IBM. Consultado em 8 de julho de 2008

## Ver também

- Depurador
- Programação
- Teste de software

Obtida de "https://pt.wikipedia.org/w/index.php?title=Depuração&oldid=53189985"

**Esta página foi editada pela última vez às 01h22min de 23 de setembro de 2018.**

Este texto é disponibilizado nos termos da licença Atribuição-Compartilhual 3.0 Não Adaptada (CC BY-SA 3.0) da Creative Commons; pode estar sujeito a condições adicionais. Para mais detalhes, consulte as condições de utilização.