FREE eBook

LEARNING debugging

Free unaffiliated eBook created from
Stack Overflow contributors.

#debugging

Table of Contents

About 1

Chapter 1: Getting started with debugging 2

Remarks 2

Examples 2

Example debugging tool chain for user mode debugging on Windows 2

Chapter 2: Understanding System Error/Exit Codes in Windows 3

Remarks 3

Examples 3

Converting exit codes into meaningful messages 3

Credits 4

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: debugging

It is an unofficial and free debugging ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official debugging.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with debugging

Remarks

This section provides an overview of what debugging is, and why a developer might want to use it.

It should also mention any large subjects within debugging, and link out to the related topics. Since the Documentation for debugging is new, you may need to create initial versions of those related topics.

Examples

Example debugging tool chain for user mode debugging on Windows

- Visual Studio (IDE)
- WPA (performance analyzer)
- WinDbg (debugger)
- IDA Pro (disassembler)
- dotPeek (decompiler for .NET)
- WinMerge (diff tool)
- HxD or 010 editor (hex editor)
- Speedcrunch (calculator)
- Firefox (browser)
- Rohitab API monitor (API call monitoring)
- SOS (a WinDbg extension for .NET)
- SOSex (another extension for WinDbg)
- Agent Ransack (file content search)
- WSCC (collection of SysInternals and Nirsoft tools)
- Debug Diag (diagnosis tool)
- Application verifier (runtime analysis tool)
- Dependency walker (DLL dependency analysis tool)
- Stud_PE (PE file analysis tool)

Read Getting started with debugging online: https://riptutorial.com/debugging/topic/5312/getting-started-with-debugging

Chapter 2: Understanding System Error/Exit Codes in Windows

Remarks

For this process to work, the original error/exit code should start with `0x8007` which generally is an indication it originated from a valid Win32 process.

However, should no message appear, then it probably didn't originate from a Windows process and therefore, will need to be examined further outside the steps mentioned above.

Examples

Converting exit codes into meaningful messages

Being able to understand Error/Exit codes is a fundamental skill for developers on Window's machine. Yet for many, the cryptic hexadecimal code that can be produced on an application exiting with on error can prove to be time consuming and painstaking process for the developer to track down and isolate.

For instance, on SO, there are several thousand questions all asking about the meaning of what a particular error/exit code means... and as an example, below is one such exit code

> The program '[4432] program.exe' has exited with code -2147023895 (0x800703e9).

Therefore, in order to identify the cause of the issue we need to convert the exit/error code into something more meaningful and we can do this by undertaking the following process.

1. From the error code `0x800703e9`, take the last 4 characters `03e9`
2. Using a Hexadecimal to Decimal Converter, convert `03e9` to its decimal counterpart, which in this case is `1001`
3. Using `cmd`, type `net helpmsg 1001` or whatever decimal value is returned from step 3.
4. A friendly error message should appear that can help identify the cause of the issue, which in this instance, the error returned was `Recursion too deep; the stack overflowed`.

Read Understanding System Error/Exit Codes in Windows online: https://riptutorial.com/debugging/topic/6146/understanding-system-error-exit-codes-in-windows

Credits

S. No	Chapters	Contributors
1	Getting started with debugging	Community, Daniel Nugent, Thomas Weller
2	Understanding System Error/Exit Codes in Windows	timkly