Version
Visual Studio 2022
Filter by title
_CrtDumpMemoryLeaks
_CrtGetAllocHook
_CrtGetDumpClient
_CrtGetReportHook
_CrtIsMemoryBlock
_CrtIsValidHeapPointer
_CrtIsValidPointer
_CrtMemCheckpoint
_CrtMemDifference
_CrtMemDumpAllObjectsSince
_CrtMemDumpStatistics
_CrtSetDebugFillThreshold
_CrtSetDumpClient
_CrtSetReportFile
_CrtSetReportHook
_CrtSetReportHook2, _CrtSetReportHookW2
_CrtSetReportMode
cscanf
_cscanf, _cscanf_l, _cwscanf, _cwscanf_l
_cscanf_s, _cscanf_s_l, _cwscanf_s, _cwscanf_s_l
csin, csinf, csinl
csinh, csinhf, csinhl
csqrt, csqrtf, csqrtl
ctan, ctanf, ctanl
ctanh, ctanhf, ctanhl
ctime, _ctime32, _ctime64, _wctime, _wctime32, _wctime64
ctime_s, _ctime32_s, _ctime64_s, _wctime_s, _wctime32_s, _wctime64_s
cwait
_cwait
_CxxThrowException
difftime, _difftime32, _difftime64
div
dup, dup2
_dup, _dup2
_dupenv_s, _wdupenv_s
_dupenv_s_dbg, _wdupenv_s_dbg
ecvt
_ecvt
_ecvt_s
_endthread, _endthreadex
eof
_eof
erf, erff, erfl, erfc, erfcf, erfcl
execl
_execl, _wexecl
execle
_execle, _wexecle
execlp
_execlp, _wexeclp
execlpe
_execlpe, _wexeclpe
execv
_execv, _wexecv
execve
_execve, _wexecve
execvp
_execvp, _wexecvp
execvpe
_execvpe, _wexecvpe
exit, _Exit, _exit
exp, expf, expl
exp2, exp2f, exp2l
_expand
_expand_dbg
expm1, expm1f, expm1l
fabs, fabsf, fabsl
fclose, _fcloseall
_fclose_nolock
fcloseall
fcvt
_fcvt
_fcvt_s
fdim, fdimf, fdiml
fdopen
_fdopen, _wfdopen
feclearexcept
fegetenv
fegetexceptflag
Download PDF

_CrtDumpMemoryLeaks

Article • 02/12/2022 • 2 minutes to read • 9 contributors

Dumps all the memory blocks in the debug heap when a memory leak has occurred (debug version only).

Syntax

```
C
int _CrtDumpMemoryLeaks( void );
```

Return Value

`_CrtDumpMemoryLeaks` returns `TRUE` if a memory leak is found. Otherwise, the function returns `FALSE`.

Remarks

The `_CrtDumpMemoryLeaks` function determines whether a memory leak has occurred since the start of program execution. When a leak is found, the debug header information for all the objects in the heap is dumped in a user-readable form. When _DEBUG isn't defined, calls to `_CrtDumpMemoryLeaks` are removed during preprocessing.

`_CrtDumpMemoryLeaks` is frequently called at the end of program execution to verify that all memory allocated by the application has been freed. The function can be called automatically at program termination by turning on the `_CRTDBG_LEAK_CHECK_DF` bit field of the _crtDbgFlag flag using the _CrtSetDbgFlag function.

`_CrtDumpMemoryLeaks` calls _CrtMemCheckpoint to obtain the current state of the heap and then scans the state for blocks that haven't been freed. When an unfreed block is encountered, `_CrtDumpMemoryLeaks` calls _CrtMemDumpAllObjectsSince to dump information for all the objects allocated in the heap from the start of program execution.

By default, internal C run-time blocks (`_CRT_BLOCK`) aren't included in memory dump operations. The _CrtSetDbgFlag function can be used to turn on the `_CRTDBG_CHECK_CRT_DF` bit of `_crtDbgFlag` to include these blocks in the leak detection process.

For more information about heap state functions and the `_CrtMemState` structure, see Heap State Reporting Functions. For more information about how memory blocks are allocated, initialized, and managed in the debug version of the base heap, see CRT Debug Heap Details.

Requirements

Routine	Required header
`_CrtDumpMemoryLeaks`	`<crtdbg.h>`

For more compatibility information, see Compatibility.

Libraries

Debug versions of C run-time libraries only.

Example

For a sample of how to use `_CrtDumpMemoryLeaks`, see crt_dbg1.

See also

Debug Routines

Recommended content

__cdecl
Learn more about: __cdecl

pack pragma
Learn more about the pack pragma directive in Microsoft C/C++

CRT Debug Heap Details - Visual Studio (Windows)
The debug heap provides powerful tools to help solve memory allocation problems. Learn about the tools and how they help with problems such as leaks and overruns.

Security Features in the CRT
An overview of secure CRT functions in the Microsoft C runtime.

Show more

Feedback

Submit and view feedback for

This product | This page

View all page feedback

In this article
Syntax
Return Value
Remarks
Requirements
Libraries
Example
See also