In this article

Return Value

Requirements

Syntax

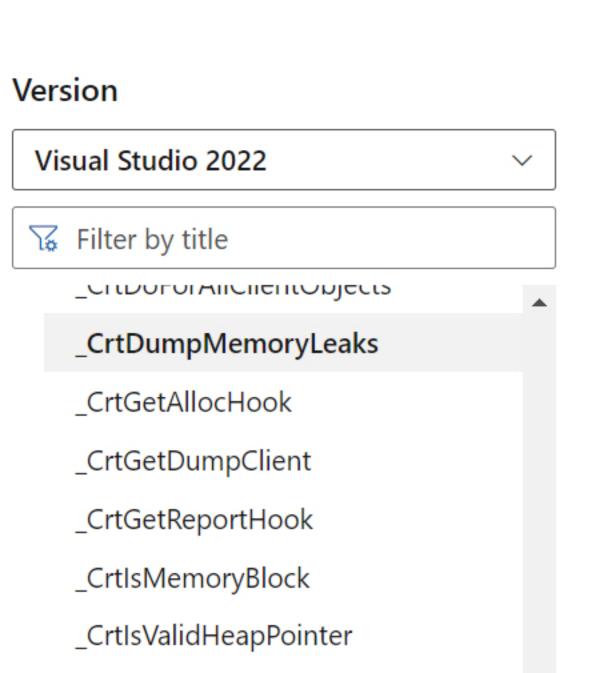
Remarks

Libraries

Example

See also

**△ ▽** 



\_CrtDumpMemoryLeaks

\_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



#### 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\_вьоск) aren't included in memory dump operations. The \_CrtSetDbgFlag function can be used to turn on the <u>crtdbg\_check\_crt\_df</u> bit of <u>crtdbgflag</u> 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></crtdbg.h>

For more compatibility information, see Compatibility.

#### Libraries

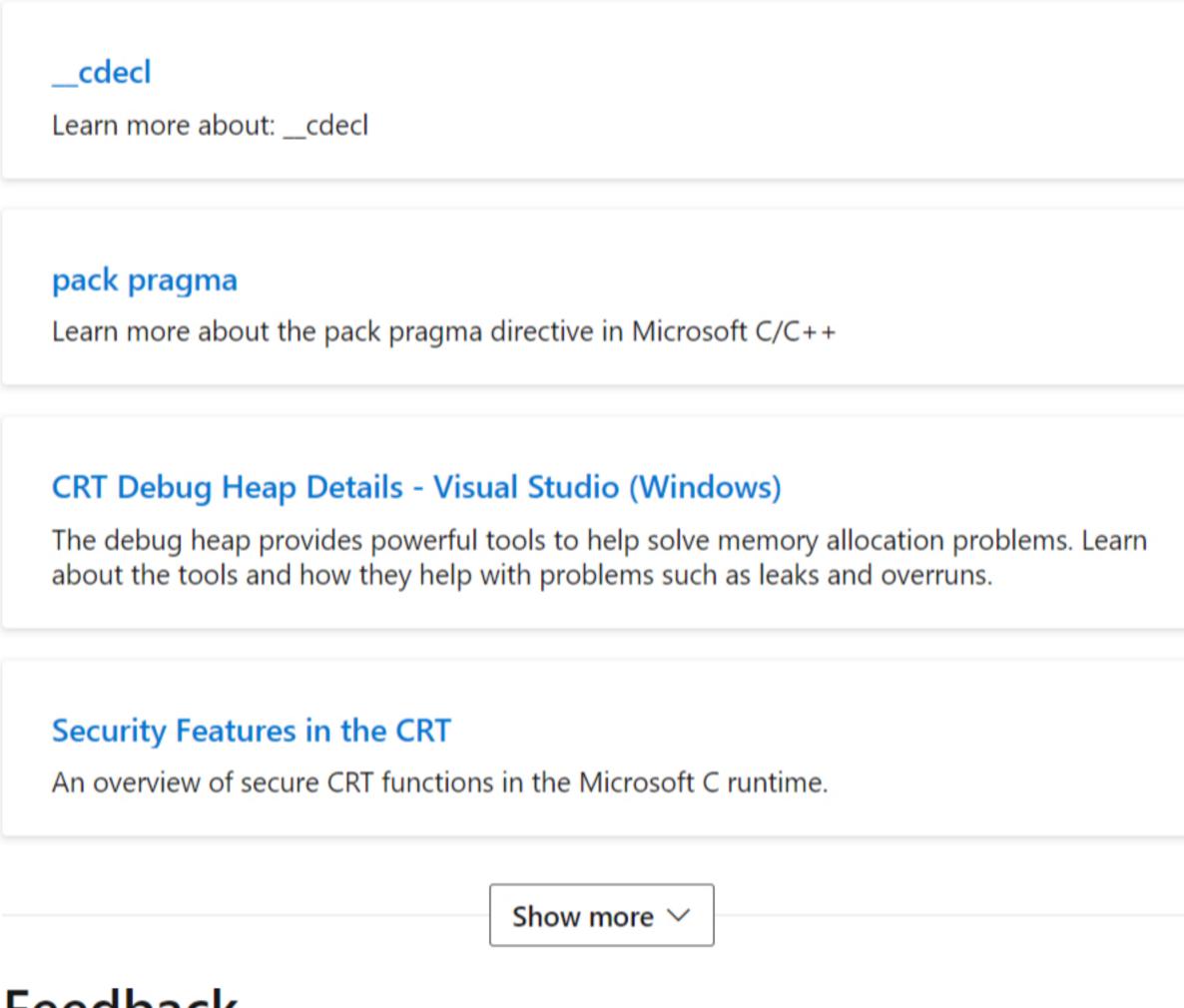
Debug versions of C run-time libraries only.

## Example

#### See also

**Debug Routines** 

### Recommended content



#### Feedback



View all page feedback ☑

\_CrtlsValidPointer \_CrtMemCheckpoint

> \_CrtMemDumpAllObjectsSince \_CrtMemDumpStatistics \_CrtSetDebugFillThreshold

\_CrtSetDumpClient \_CrtSetReportFile

\_CrtMemDifference

\_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

execl

erf, erff, erfl, erfc, erfcf, erfcl

\_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

Download PDF

fegetexceptflag

States) English (United States)

Blog