**常规内存函数**

| **函数** | **说明** |
| --- | --- |
| [**AddSecureMemoryCacheCallback**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-addsecurememorycachecallback) | 注册在释放安全内存范围或更改其保护时调用的回调函数。 |
| [CopyDeviceMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-copydevicememory) | 在开发人员需要额外确保访问设备内存时不会产生对齐错误的情况下，将内存从一个位置复制到另一个位置，且不会受到编译器优化的干扰。 |
| [**CopyMemory**](https://learn.microsoft.com/zh-cn/previous-versions/windows/desktop/legacy/aa366535(v=vs.85)) | 将内存块从一个位置复制到另一个位置。 |
| [CopyVolatileMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-copyvolatilememory) | 将源内存块的内容复制到目标内存块。 |
| [**CreateMemoryResourceNotification**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-creatememoryresourcenotification) | 创建内存资源通知对象。 |
| [FillDeviceMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-filldevicememory) | 在开发人员需要额外确保访问设备内存时不会产生对齐错误的情况下，设置缓冲区的内容，且不会受到编译器优化的干扰。 |
| [**FillMemory**](https://learn.microsoft.com/zh-cn/previous-versions/windows/desktop/legacy/aa366561(v=vs.85)) | 用指定的值填充内存块。 |
| [FillVolatileMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-fillvolatilememory) | 使用指定的填充值填充内存块。 |
| [**GetLargePageMinimum**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-getlargepageminimum) | 检索大型页面的最小大小。 |
| [**GetPhysicallyInstalledSystemMemory**](https://learn.microsoft.com/zh-cn/windows/win32/api/sysinfoapi/nf-sysinfoapi-getphysicallyinstalledsystemmemory) | 检索计算机上物理安装的 RAM 容量。 |
| [**GetSystemFileCacheSize**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-getsystemfilecachesize) | 检索系统缓存的工作集的当前大小限制。 |
| [**GetWriteWatch**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-getwritewatch) | 检索已写入虚拟内存区域中的页面的地址。 |
| [**GlobalMemoryStatusEx**](https://learn.microsoft.com/zh-cn/windows/win32/api/sysinfoapi/nf-sysinfoapi-globalmemorystatusex) | 获取有关系统当前使用物理内存和虚拟内存的信息。 |
| [**MoveMemory**](https://learn.microsoft.com/zh-cn/previous-versions/windows/desktop/legacy/aa366788(v=vs.85)) | 将内存块从一个位置移到另一个位置。 |
| [MoveVolatileMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-movevolatilememory) | 将源内存块的内容复制到目标内存块，并支持重叠源内存块和目标内存块。 |
| [**QueryMemoryResourceNotification**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-querymemoryresourcenotification) | 检索指定内存资源对象的状态。 |
| [**RemoveSecureMemoryCacheCallback**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-removesecurememorycachecallback) | 注销以前向 [**AddSecureMemoryCacheCallback**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-addsecurememorycachecallback) 函数注册的回调函数。 |
| [**ResetWriteWatch**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-resetwritewatch) | 重置虚拟内存区域的写入跟踪状态。 |
| [**SecureMemoryCacheCallback**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinNT/nc-winnt-psecure_memory_cache_callback) | 在释放安全内存范围或更改其保护时调用的应用程序定义函数。 |
| [**SecureZeroMemory**](https://learn.microsoft.com/zh-cn/previous-versions/windows/desktop/legacy/aa366877(v=vs.85)) | 用零填充内存块。 |
| [SecureZeroMemory2](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-securezeromemory2) | 以保证安全的方式用零填充内存块。 |
| [**SetSystemFileCacheSize**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-setsystemfilecachesize) | 限制文件系统缓存的工作集的大小。 |
| [ZeroDeviceMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-zerodevicememory) | 在开发人员需要额外确保访问设备内存时不会产生对齐错误的情况下，将缓冲区的内容设置为零，且不会受到编译器优化的干扰。 |
| [**ZeroMemory**](https://learn.microsoft.com/zh-cn/previous-versions/windows/desktop/legacy/aa366920(v=vs.85)) | 用零填充内存块。 |
| [ZeroVolatileMemory](https://learn.microsoft.com/zh-cn/windows/win32/memory/winbase-zerovolatilememory) | 用零填充内存块。 |

**数据执行防护函数**

这些函数用于[数据执行防护](https://learn.microsoft.com/zh-cn/windows/win32/memory/data-execution-prevention) (DEP)。

展开表

| **函数** | **说明** |
| --- | --- |
| [**GetProcessDEPPolicy**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-getprocessdeppolicy) | 检索进程的 DEP 设置。 |
| [**GetSystemDEPPolicy**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-getsystemdeppolicy) | 检索系统的 DEP 设置。 |
| [**SetProcessDEPPolicy**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-setprocessdeppolicy) | 更改进程的 DEP 设置。 |

**文件映射函数**

这些函数用于[文件映射](https://learn.microsoft.com/zh-cn/windows/win32/memory/file-mapping)。

展开表

| **函数** | **说明** |
| --- | --- |
| [**CreateFileMappingA**](https://learn.microsoft.com/zh-cn/windows/win32/api/winbase/nf-winbase-createfilemappinga) | 为指定的文件创建或打开命名或未命名的文件映射对象。 |
| [**CreateFileMappingW**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-createfilemappingw) | 为指定的文件创建或打开命名或未命名的文件映射对象。 |
| [**CreateFileMapping2**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-createfilemapping2) | 为指定的文件创建或打开命名或未命名的文件映射对象。 可以将物理内存的首选 NUMA 节点指定为扩展参数；请参阅 *ExtendedParameters* 参数。 |
| [**CreateFileMappingFromApp**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-createfilemappingfromapp) | 为 Windows 应用商店应用中指定的文件创建或打开命名或未命名的文件映射对象。 |
| [**CreateFileMappingNuma**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-createfilemappingnumaa) | 为指定的文件创建或打开命名或未命名的文件映射对象，并为物理内存指定 NUMA 节点。 |
| [**FlushViewOfFile**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-flushviewoffile) | 在文件的映射视图中写入磁盘字节范围。 |
| [**GetMappedFileName**](https://learn.microsoft.com/zh-cn/windows/win32/api/psapi/nf-psapi-getmappedfilenamea) | 检查指定的地址是否位于指定进程的地址空间的内存映射文件中。 如果是，该函数将返回内存映射文件的名称。 |
| [**MapViewOfFile**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-mapviewoffile) | 将文件映射的视图映射到调用进程的地址空间。 |
| [**MapViewOfFile2**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-mapviewoffile2) | 将文件或页面文件支持的分区视图映射到指定进程的地址空间。 |
| [**MapViewOfFile3**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-mapviewoffile3) | 将文件或页面文件支持的分区视图映射到指定进程的地址空间。 |
| [**MapViewOfFile3FromApp**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-mapviewoffile3fromapp) | 将 Windows 商店应用中文件映射的视图映射到调用进程的地址空间。 |
| [**MapViewOfFileEx**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-mapviewoffileex) | 将文件映射的视图映射到调用进程的地址空间。 调用方可以选择为视图指定建议的内存地址。 |
| [**MapViewOfFileExNuma**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-mapviewoffileexnuma) | 将文件映射的视图映射到调用进程的地址空间，并为物理内存指定 NUMA 节点。 |
| [**MapViewOfFileFromApp**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-mapviewoffilefromapp) | 将 Windows 商店应用中文件映射的视图映射到调用进程的地址空间。 |
| [**MapViewOfFileNuma2**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-mapviewoffilenuma2) | 将文件或页面文件支持的分区视图映射到指定进程的地址空间。 |
| [**OpenFileMapping**](https://learn.microsoft.com/zh-cn/windows/win32/api/winbase/nf-winbase-openfilemappinga) | 打开命名的文件映射对象。 |
| [**OpenFileMappingFromApp**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-openfilemappingfromapp) | 打开命名的文件映射对象。 |
| [**UnmapViewOfFile**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-unmapviewoffile) | 从调用进程的地址空间取消映射文件的映射视图。 |
| [**UnmapViewOfFile2**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-unmapviewoffile2) | 取消映射以前映射的文件视图或页面文件支持的分区。 |
| [**UnmapViewOfFileEx**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-unmapviewoffileex) | 取消映射以前映射的文件视图或页面文件支持的分区。 |

**AWE 函数**

以下是 [AWE 函数](https://learn.microsoft.com/zh-cn/windows/win32/memory/address-windowing-extensions)。

展开表

| **函数** | **说明** |
| --- | --- |
| [**AllocateUserPhysicalPages**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-allocateuserphysicalpages) | 在进程的任何 AWE 区域中分配要映射和取消映射的物理内存页。 |
| [**AllocateUserPhysicalPagesNuma**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-allocateuserphysicalpagesnuma) | 在进程的任何 AWE 区域中分配要映射和取消映射的物理内存页，并为物理内存指定 NUMA 节点。 |
| [**FreeUserPhysicalPages**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-freeuserphysicalpages) | 释放以前使用 [**AllocateUserPhysicalPages**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-allocateuserphysicalpages) 分配的物理内存页。 |
| [**MapUserPhysicalPages**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-mapuserphysicalpages) | 映射以前在 AWE 区域中的指定地址分配的物理内存页。 |
| [**MapUserPhysicalPagesScatter**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-mapuserphysicalpagesscatter) | 映射以前在 AWE 区域中的指定地址分配的物理内存页。 |

**堆函数**

以下是[堆函数](https://learn.microsoft.com/zh-cn/windows/win32/memory/heap-functions)。

展开表

| **函数** | **说明** |
| --- | --- |
| [**GetProcessHeap**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-getprocessheap) | 获取调用进程的堆的句柄。 |
| [**GetProcessHeaps**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-getprocessheaps) | 获取对调用进程有效的所有堆的句柄。 |
| [**HeapAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapalloc) | 从堆中分配内存块。 |
| [**HeapCompact**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapcompact) | 合并堆上的相邻可用内存块。 |
| [**HeapCreate**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapcreate) | 创建堆对象。 |
| [**HeapDestroy**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapdestroy) | 销毁指定的堆对象。 |
| [**HeapFree**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapfree) | 释放从堆分配的内存块。 |
| [**HeapLock**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heaplock) | 尝试获取与指定堆关联的锁。 |
| [**HeapQueryInformation**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapqueryinformation) | 检索有关指定的堆的信息。 |
| [**HeapReAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heaprealloc) | 从堆重新分配内存块。 |
| [**HeapSetInformation**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapsetinformation) | 设置指定堆的堆信息。 |
| [**HeapSize**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapsize) | 检索从堆分配的内存块的大小。 |
| [**HeapUnlock**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapunlock) | 释放与指定堆关联的锁定的所有权。 |
| [**HeapValidate**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapvalidate) | 尝试验证指定的堆。 |
| [**HeapWalk**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapwalk) | 枚举指定堆中的内存块。 |

**虚拟内存函数**

以下是[虚拟内存函数](https://learn.microsoft.com/zh-cn/windows/win32/memory/virtual-memory-functions)。

展开表

| **函数** | **说明** |
| --- | --- |
| [**DiscardVirtualMemory**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-discardvirtualmemory) | 放弃一系列内存页面的内存内容，而不会解除内存的使用。 已放弃内存的内容未定义，必须由应用程序重写。 |
| [**OfferVirtualMemory**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-offervirtualmemory) | 指示应用程序不再需要一系列内存页中包含的数据，并可以在必要时被系统放弃。 |
| [**PrefetchVirtualMemory**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-prefetchvirtualmemory) | 将虚拟地址范围预提取到物理内存中。 |
| [**QueryVirtualMemoryInformation**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-queryvirtualmemoryinformation) | 返回有关指定进程的虚拟地址空间中的一个或一组页面的信息。 |
| [**ReclaimVirtualMemory**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-reclaimvirtualmemory) | 使用 [**OfferVirtualMemory**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-offervirtualmemory) 回收系统提供的一系列内存页面。 |
| [**SetProcessValidCallTargets**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-setprocessvalidcalltargets) | 为 CFG 提供有效的间接调用目标列表，并指定它们是否应标记为有效。 |
| [**VirtualAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-virtualalloc) | 保留或提交调用进程的虚拟地址空间中的页面区域。 |
| [**VirtualAlloc2**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-virtualalloc2) | 保留、提交或更改指定进程的虚拟地址空间中内存区域的状态。 该函数初始化它分配给零的内存。 |
| [**VirtualAlloc2FromApp**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-virtualallocfromapp) | 保留、提交或更改调用进程的虚拟地址空间中页范围的状态。 此函数分配的内存会自动初始化为零。 |
| [**VirtualAllocEx**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualallocex) | 保留或提交指定进程的虚拟地址空间中的页面区域。 |
| [**VirtualAllocExNuma**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualallocexnuma) | 保留或提交指定进程的虚拟地址空间中的内存区域，并为物理内存指定 NUMA 节点。 |
| [**VirtualAllocFromApp**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-virtualallocfromapp) | 保留、提交或更改调用进程的虚拟地址空间中页范围的状态。 此函数分配的内存会自动初始化为零。 |
| [**VirtualFree**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualfree) | 释放或取消提交调用进程的虚拟地址空间中的页面区域。 |
| [**VirtualFreeEx**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualfreeex) | 释放或解除指定进程的虚拟地址空间中的内存区域。 |
| [**VirtualLock**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtuallock) | 将进程的虚拟地址空间的指定区域锁定到物理内存中。 |
| [**VirtualProtect**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualprotect) | 更改调用进程的虚拟地址空间中已提交的页面区域的访问保护。 |
| [**VirtualProtectEx**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualprotectex) | 更改调用进程的虚拟地址空间中已提交的页面区域的访问保护。 |
| [**VirtualProtectFromApp**](https://learn.microsoft.com/zh-cn/windows/desktop/api/MemoryApi/nf-memoryapi-virtualprotectfromapp) | 更改调用进程的虚拟地址空间中已提交的页面区域的保护。 |
| [**VirtualQuery**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualquery) | 提供有关调用进程的虚拟地址空间中一系列页面的信息。 |
| [**VirtualQueryEx**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualqueryex) | 提供有关调用进程的虚拟地址空间中一系列页面的信息。 |
| [**VirtualUnlock**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-virtualunlock) | 解锁进程的虚拟地址空间中的指定页面范围。 |

**全局和本地函数**

另请参阅[全局和本地函数](https://learn.microsoft.com/zh-cn/windows/win32/memory/global-and-local-functions)。 提供这些函数是为了与 16 位 Windows 兼容，并与动态数据交换 (DDE)、剪贴板函数和 OLE 数据对象一起使用。 除非文档特别指出应使用全局函数或本地函数，否则新应用程序应使用具有 [**GetProcessHeap**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-getprocessheap) 返回的句柄的相应[堆函数](https://learn.microsoft.com/zh-cn/windows/win32/memory/heap-functions)。 对于与全局函数或本地函数等效的功能，请将堆函数的 *dwFlags* 参数设置为 0。

展开表

| **函数** | **说明** | **相应的堆函数** |
| --- | --- | --- |
| [**GlobalAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalalloc)、[**LocalAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localalloc) | 从堆中分配指定的字节数。 | [**HeapAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapalloc) |
| [**GlobalDiscard**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globaldiscard)、[**LocalDiscard**](https://learn.microsoft.com/zh-cn/windows/win32/api/minwinbase/nf-minwinbase-localdiscard) | 放弃指定的全局内存块。 | 不适用。 |
| [**GlobalFlags**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalflags)、[**LocalFlags**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localflags) | 返回有关指定全局内存对象的信息。 | 不适用。 使用 [**HeapValidate**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapvalidate) 验证堆。 |
| [**GlobalFree**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalfree)、[**LocalFree**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localfree) | 释放指定的全局内存对象。 | [**HeapFree**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapfree) |
| [**GlobalHandle**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalhandle)、[**LocalHandle**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localhandle) | 检索与指向全局内存块的指定指针关联的句柄。 此函数应仅用于需要它的 OLE 和剪贴板函数。 | 不适用。 |
| [**GlobalLock**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globallock)、[**LocalLock**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-locallock) | 锁定全局内存对象，并返回指向对象内存块的第一个字节的指针。 | 不适用。 |
| [**GlobalReAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalrealloc)、[**LocalReAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localrealloc) | 更改指定全局内存对象的大小或属性。 | [**HeapReAlloc**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heaprealloc) |
| [**GlobalSize**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalsize)、[**LocalSize**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localsize) | 检索指定全局内存对象的当前大小。 | [**HeapSize**](https://learn.microsoft.com/zh-cn/windows/desktop/api/HeapApi/nf-heapapi-heapsize) |
| [**GlobalUnlock**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-globalunlock)、[**LocalUnlock**](https://learn.microsoft.com/zh-cn/windows/desktop/api/WinBase/nf-winbase-localunlock) | 递减与内存对象关联的锁定计数。 此函数应仅用于需要它的 OLE 和剪贴板函数。 | 不适用。 |

**错误内存函数**

展开表

| **函数** | **说明** |
| --- | --- |
| [**BadMemoryCallbackRoutine**](https://learn.microsoft.com/zh-cn/previous-versions/windows/desktop/legacy/hh691011(v=vs.85)) | 向 [**RegisterBadMemoryNotification**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-registerbadmemorynotification) 函数注册的应用程序定义函数，在检测到一个或多个错误的内存页面时调用。 |
| [**GetMemoryErrorHandlingCapabilities**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-getmemoryerrorhandlingcapabilities) | 获取系统的内存错误处理功能。 |
| [**RegisterBadMemoryNotification**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-registerbadmemorynotification) | 注册检测到一个或多个错误内存页面时调用的内存错误通知。 |
| [**UnregisterBadMemoryNotification**](https://learn.microsoft.com/zh-cn/windows/win32/api/memoryapi/nf-memoryapi-unregisterbadmemorynotification) | 关闭指定的错误内存通知句柄。 |

**Enclave 函数**

展开表

| **函数** | **说明** |
| --- | --- |
| [**CreateEnclave**](https://learn.microsoft.com/zh-cn/windows/desktop/api/enclaveapi/nf-enclaveapi-createenclave) | 创建新的未初始化 enclave。 enclave 是应用程序地址空间内代码和数据的独立区域。 只有 enclave 中运行的代码才能访问同一 enclave 中的数据。 |
| [**InitializeEnclave**](https://learn.microsoft.com/zh-cn/windows/desktop/api/enclaveapi/nf-enclaveapi-initializeenclave) | 初始化使用数据创建和加载的 enclave。 |
| [**IsEnclaveTypeSupported**](https://learn.microsoft.com/zh-cn/windows/desktop/api/enclaveapi/nf-enclaveapi-isenclavetypesupported) | 检索是否支持指定的 enclave 类型。 |
| [**LoadEnclaveData**](https://learn.microsoft.com/zh-cn/windows/desktop/api/enclaveapi/nf-enclaveapi-loadenclavedata) | 将数据加载到通过调用 [**CreateEnclave**](https://learn.microsoft.com/zh-cn/windows/desktop/api/enclaveapi/nf-enclaveapi-createenclave) 创建的未初始化 enclave。 |

**ATL thunk 函数**

展开表

| **函数** | **说明** |
| --- | --- |
| [**AtlThunk\_AllocateData**](https://learn.microsoft.com/zh-cn/windows/desktop/api/atlthunk/nf-atlthunk-atlthunk_allocatedata) | 为 ATL thunk 分配内存中的空间。 |
| [**AtlThunk\_DataToCode**](https://learn.microsoft.com/zh-cn/windows/desktop/api/atlthunk/nf-atlthunk-atlthunk_datatocode) | 返回对应于 AtlThunkData\_t 参数的可执行函数。 |
| [**AtlThunk\_FreeData**](https://learn.microsoft.com/zh-cn/windows/desktop/api/atlthunk/nf-atlthunk-atlthunk_freedata) | 释放与 ATL thunk 关联的内存。 |
| [**AtlThunk\_InitData**](https://learn.microsoft.com/zh-cn/windows/desktop/api/atlthunk/nf-atlthunk-atlthunk_initdata) | 初始化 ATL thunk。 |