

# Index

- A**
- Abandoned mutexes 247
  - AbnormalTermination function 104
  - accept function 351
  - Access
    - rights 434
    - tokens 432, 454
  - Access
    - for read and write 27
    - word 137, 172
  - Access control entries (ACEs) 434, 438–439, 445–447, 453
  - Access control lists (ACLs) 434, 438–439, 445–447, 453
  - Access control lists, discretionary (DACLS) 433
  - access UNIX function 44
  - ACCESS\_ALLOWED\_ACE 447
  - ACCESS\_DENIED\_ACE 447
  - Account pointer 436
  - AceCount 447
  - AceType 447
  - ACL\_REVISION word 438
  - ACL\_SIZE\_INFORMATION 447
  - AclBytesFree 447
  - AclBytesInUse 447
  - AclInfoClass 447
  - AclInformation 447
  - AclRevisionInformation 447
  - AclSizeInformation 447
  - AddAccessAllowedAce function 438
  - AddAccessDeniedAce function 438
  - AddAuditAccessAce function 453
  - Address space 120
  - AddSharedElement 248
  - AddVectoredExceptionHandler 115
  - AF\_INET 348
  - Alertable
    - I/O 411
    - wait functions 413–415
  - Alertable flag 288, 414
  - AllocateAndInitializesid
    - function 437, 454
  - AllocConsole function 38
  - Anonymous pipes 316
  - Apartment model threading 342
  - Arguments array for exceptions 100
  - Asc2Un ASCII to Unicode program 45
  - Asc2UnMM program 141
  - Asynchronous I/O 402
    - with threads 418
  - atou
    - file conversion program 45
    - performance 510
  - atouEL program 155
  - atouEX program 416
  - atouMM performance 510
  - atouOV
    - performance 510
    - program 408
  - Attributes
    - directory 65–67
    - file 63–67
  - AttrsAndFlags word 28, 403
- B**
- \_based keyword 145
  - Based pointers 144
  - \_beginthreadex Microsoft C
    - function 207–208
  - Berkeley Sockets 346, 374
  - Binary search tree 129–130
  - BinaryPathName pointer 392
  - bind function 349
  - Boss/worker model 211–212
  - Broadcast mechanisms 335
  - Buffer pointer 30
  - Bytes heap size 127
- C**
- C library 12
    - in threads 207–208
  - C2 security 431

- CallNamedPipe function 326
- calloc C library function 129
- CancelWaitableTimer function 420
- cat
  - program 43
  - UNIX command 43
  - utility 179
- CD-ROM File System (CDFS) 24
- ChangeFilePermission program 449
- ChangeFilePermissionprogram 451
- ChangeServiceConfig parameter 392
- ChangeServiceConfig2 parameter 392
- CHAR type 32
- Character types 32–34
- CheckPoint word 384
- chmod
  - program 440
  - UNIX command 439
- Class key type 83
- \_clearfp function 98
- Client connections to named pipes 323
- Client/server
  - command line processor 328–335
  - model 211, 319
  - named pipe connection 324
- clientNP program 329
- clientsK program 357
- Clientsock 353
- clntsrvr.h 329, 483
- closedir UNIX function 67
- CloseHandle function 18, 29, 65, 136
- CloseServiceHandle function 392
- closesocket function 351
- CloseThresholdBarrier function 291
- Closing files 29–30
- COM 149
  - threading models 341
- command program 364
- CommandLine argument 166, 168–169
- CommandLineToArgvW Win32
  - function 488
- CompareFileTime function 66
- Comparesid function 454
- Compatibility libraries 8
- Completion routines 411–415
- Component object model
  - technology 341
- Condition variable model 282–287
- CONIN\$ pathname 37
- connect function 352
- ConnectNamedPipe
  - function 324, 403
- CONOUT\$ pathname 37
- Console
  - control events 185–186
  - control handlers 112–113, 167
  - I/O 37–39
- Console handle 37
- ConsolePrompt function 39
- Control word 387
- ControlCode word 393
- \_controlfp function 98
- ControlsAccepted word 386
- ControlService function 393
- CopyFile function 18, 47
- Copying files 47–48
- CopySid function 454
- Co-routines 224
- cp UNIX command 14
- cpc
  - C library program 14
  - performance 508
- cpcf
  - performance 508
  - Win32 program 18
- cpuc performance 508
- cpw
  - performance 508
  - Win32 file copying program 16
- cpwfa performance 508
- Create word 27, 166–167, 203
- CREATE\_ALWAYS flag 27
- CREATE\_NEW flag 27
- CREATE\_NEW\_CONSOLE flag 166
- CREATE\_NEW\_PROCESS\_GROUP flag 167, 185
- CREATE\_SUSPENDED flag 166, 203
- CreateDirectory function 49
- CreateEvent function 253, 405
- CreateFile flag 59
- CreateFile function 26–29, 64, 403

- CreateFileMapping function 135–136
- CreateHardLink 47, 65
- CreateIndexFile function 147
- CreateIoCompletionPort function 423
- CreateMailslot function 337–338
- CreateMutex function 246
- CreateNamedPipe function 321, 403
- CreatePipe function 316
- CreatePrivateObjectSecurity function 451
- CreateProcess function 166–168, 185, 218
- CreateRemoteThread function 206
- CreateSemaphore function 250
- CreateThread function 202–203
- CreateThresholdBarrier function 290
- CreateWaitableTimer function 419
- Creating
  - directories 49–50
  - files 26–29
- Critical sections 236–249, 259–260
- CRITICAL\_SECTION 282
- CrtlType word 113
- CTRL\_BREAK\_EVENT 185
- CTRL\_BREAK\_EVENT flag 113
- CTRL\_C\_EVENT flag 113, 185
- CTRL\_CLOSE\_EVENT flag 113
- CTRL\_LOGOFF\_EVENT flag 113
- CTRL\_SHUTDOWN\_EVENT flag 113
- ctrlc signal handling program 114
- CurDir
  - argument 167
  - path string 50
- CurrentState word 385
- D**
- DACL\_SECURITY\_INFORMATION 446
- Dac1Defaulted flag 439
- Dac1Present flag 439
- Data types 462–463
- DatabaseName pointer 391
- Datagrams 373–374
- Deadlocks 247–249
- Debugging 309–311
- \_declspec C++ storage modifier 151
- DeleteAce function 453
- DeleteFile function 47
- DeleteMountPoint 69
- DeleteService function 392
- DeleteSharedElement 248
- Deleting
  - directories 49–50
  - files 47–48
- DeregisterEventSource 398
- Design 309–311
- Desired security mask for keys 81
- DesiredAccess word 136, 391, 392
- DETACHED\_PROCESS flag 166, 186
- DeviceIoControl function 60
- Directories
  - attributes 65–67
  - creating 49–50
  - deleting 49–50
  - managing and setting 50–51
  - moving 48–49
  - naming 25, 49
  - setting 169
- DisconnectNamedPipe function 324
- DisplayName pointer 392
- DistanceToMove value 57
- DistanceToMoveHigh pointer 57
- dllexport storage modifier 151
- dllimport storage modifier 151
- DllMain function 156
- Drive names 25
- DUPLICATE\_CLOSE\_SOURCE flag 174
- DUPLICATE\_SAME\_ACCESS flag 174
- DuplicateHandle function 173
- Duplicating handles 173
- dw prefix 26
- DWORD word 26
- Dynamic
  - data structures 119
  - link libraries (DLLs) 135, 149–157
  - memory management 119–121
- E**
- EM\_DENORMAL mask 99
- EM\_INEXACT mask 99
- EM\_INVALID mask 99
- EM\_OVERFLOW mask 99
- EM\_UNDERFLOW mask 99
- EM\_ZERODIVIDE mask 99

- ENABLE\_ECHO\_INPUT flag 38
  - ENABLE\_LINE\_INPUT flag 37
  - ENABLE\_PROCESSED\_INPUT flag 38
  - ENABLE\_PROCESSED\_OUTPUT flag 38
  - ENABLE\_WRAP\_AT\_EOL\_OUTPUT flag 38
  - \_endthreadex Microsoft C function 207
  - EnterCriticalSection function 237
  - Envirmnt.h 39, 480
  - Environment block 167, 177–178
  - Environment strings 177–178
  - ERROR\_HANDLE\_EOF 413
  - ERROR\_IO\_INCOMPLETE 406
  - ERROR\_PIPE\_CONNECTED 324
  - ERROR\_SUCCESS flag 81
  - Errors 99–101
  - /etc UNIX directory 79
  - Event handle 405, 415
  - eventPC program 255
  - Events 91–92, 253–255, 282
    - logging 398
  - EvryThng.h 42, 480
  - \_\_except 92
  - Exception handlers 91–101
  - EXCEPTION\_BREAKPOINT exception
    - code 96
  - EXCEPTION\_CONTINUE\_EXECUTION 116
  - EXCEPTION\_CONTINUE\_EXECUTION
    - return value 94
  - EXCEPTION\_CONTINUE\_SEARCH 116
  - EXCEPTION\_EXECUTE\_HANDLER return
    - value 93
  - EXCEPTION\_FLT\_OVERFLOW exception
    - code 96
  - EXCEPTION\_FLT\_STACK\_CHECK exception
    - code 99
  - EXCEPTION\_INT\_DIVIDE\_BY\_ZERO
    - exception code 96
  - EXCEPTION\_MAXIMUM\_PARAMETERS 101
  - EXCEPTION\_NONCONTINUABLE exception
    - code 100
  - EXCEPTION\_NONCONTINUABLE\_EXECU-
    - TION exception code 96
  - EXCEPTION\_POINTERS 96, 116
  - EXCEPTION\_PRIV\_INSTRUCTION
    - exception code 102
  - EXCEPTION\_RECORD structure 96
  - EXCEPTION\_SINGLE\_STEP exception
    - code 96
  - ExceptionCode word 100
  - ExceptionFlags word 100
  - Exclude.h 485
  - Excption program 109
  - exec UNIX functions 168
  - execl UNIX function 168
  - Executable image 168–169
  - Existing string 48
  - ExitCode word 175, 204
  - ExitProcess function 174, 203, 206
  - ExitThread function 204
  - Explicit linking 152–154
  - Exporting and importing interfaces
    - 151–152
  - Extended I/O 411–415
- F**
- FailIfExists flag 47
  - fclose C library function 30
  - fcntl UNIX function 78
  - ferror C library function 16
  - Fibers 224–226
  - FIFO UNIX named pipe 328
  - File Allocation Table file system (FAT) 24
  - FILE C library objects 30
  - File handle 30, 57, 74, 135
  - File mapping objects 135–138
  - File permissions
    - changing 449
    - reading 447
  - FILE\_ATTRIBUTE\_DIRECTORY flag 67
  - FILE\_ATTRIBUTE\_ENCYTPED 64
  - FILE\_ATTRIBUTE\_NORMAL flag 28, 67
  - FILE\_ATTRIBUTE\_READONLY flag 28, 67
  - FILE\_ATTRIBUTE\_SPARSE\_FILE 64
  - FILE\_ATTRIBUTE\_TEMPORARY flag 67
  - FILE\_BEGIN position flag 57
  - FILE\_CURRENT position flag 57
  - FILE\_END position flag 57
  - FILE\_FLAG\_DELETE\_ON\_CLOSE flag 28
  - FILE\_FLAG\_NO\_BUFFERING 59
  - FILE\_FLAG\_NO\_BUFFERING flag 28, 403
  - FILE\_FLAG\_OVERLAPPED flag 28, 322,
    - 403, 405, 411

- FILE\_FLAG\_RANDOM\_ACCESS flag 28
- FILE\_FLAG\_SEQUENTIAL\_SCAN flag 28
- FILE\_FLAG\_WRITE\_THROUGH flag 28, 322
- FILE\_MAP\_ALL\_ACCESS word 137
- FILE\_MAP\_READ word 137
- FILE\_MAP\_WRITE word 137
- FILE\_SHARE\_READ flag 27, 338
- FILE\_SHARE\_WRITE flag 27
- FileAttributes word 64
- Files
  - attributes 63–67
  - closing 29–30
  - copying 47–48
  - creating 26–29
  - deleting 47–48
  - handles 29
  - locking 74–78
  - memory-mapped 119
  - moving 48–49
  - naming 25, 49, 67–68
  - opening 26–29
  - paging 123
  - pointers 56–58
  - reading 30–31
  - resizing 59
  - searching for 63–65
  - systems 23–24
  - writing 31
- FILETIME 66, 183, 382, 387
- FileTimeToLocalFileTime function 66
- FileTimeToSystemTime function 66
- FillTree program 132
- Filter expressions 92–94
- Filter program 111
- \_\_finally 103
- FIND\_DATA structure 64
- FindFirstFile function 59, 63–64
- FindNextFile function 64–65
- FirstHandler parameter 115
- Fixed-precision data types 462
- Floating-point exceptions 98–99
- FlushViewOfFile function 138
- fopen C library function 30
- fork UNIX function 168
- FormatMessage function 42
- fread C library function 31
- free C library function 129
- FreeConsole function 38
- FreeLibrary function 154
- freopen C library function 30
- FSCTL\_SET\_SPARSE flag 60
- ftp 352
- fwrite C library function 31
- G**
- GenerateConsoleCtrlEvent
  - function 112, 186
- Generic characters 32–34
- GENERIC\_READ 27
  - for named pipes 450
- GENERIC\_WRITE 27
  - for named pipes 450
- GetAce function 447
- GetAclInformation function 447
- GetArgs function 488
- GetCommandLine function 183
- GetCompressedFileSize function 59
- GetCurrentDirectory function 50
- GetCurrentProcess function 172
- GetCurrentProcessId function 172
- GetCurrentThread 204
- GetCurrentThreadId 204
- GetDiskFreeSpace 59
- GetEnvironmentVariable function 178
- GetExceptionCode function 95–96
- GetExceptionInformation function 96
- GetExitCodeProcess function 174–175
- GetExitCodeThread function 204
- GetFileAttributes function 66
- GetFileInformationByHandle
  - function 65
- GetFileSecurity function 446
- GetFileSize function 59
- GetFileSizeEx function 59
- GetFileType function 66
- GetFullPathName 65
- GetKernelObjectSecurity
  - function 451
- GetLastError function 18, 41
- GetMailslotInfo function 338
- GetModuleFileName function 154, 173
- GetModuleFileNameEx function 173

- GetModuleHandle function 154
- GetNamedPipeHandleState
  - function 323
- GetNamedPipeInfo function 323
- getopt UNIX function 486
- GetOverlappedResult function 406
- GetPriorityClass function 219
- GetPrivateObjectSecurity
  - function 451
- GetProcAddress function 154
- GetProcessHeap function 122, 128
- GetProcessIdOfThread function 205
- GetProcessorAffinityMask
  - function 221
- GetProcessTimes function 183
- GetQueuedCompletionStatus
  - function 424
- GetSecurityDescriptorControl
  - flag 435
- GetSecurityDescriptorDacl
  - function 446
- GetSecurityDescriptorGroup
  - function 437, 446
- GetSecurityDescriptorOwner
  - function 437, 446
- GetSecurityDescriptorSacl
  - function 453
- GetshortPathName 65
- GetStartupInfo function 167
- GetStdHandle function 36
- GetSystemDirectory function 169
- GetSystemInfo function 121
- GetTempFileName function 68
- GetTempPath function 68
- GetThreadIOPendingFlag function 205
- GetThreadPriority function 219
- GetThreadTimes function 183
- GetToken Information function 454
- GetUserName function 437
- GetVolumeNameForVolumeMountPoint
  - 69
- GetVolumePathName 69
- GetWindowsDirectory function 169
- Global storage 235
- GLOBAL\_HEAP\_SELECTED 262
- Granularity, locking 259
- grep UNIX command 179
- grepMP
  - performance 512
  - search program 180
- grepMT
  - performance 512
  - search program 209
- grepSQ performance 512
- GROUP\_SECURITY\_INFORMATION 446
- GroupDefaulted flag 437
- Growable and nongrowable heaps 124
- Guarded code blocks 92–93
- H**
- HANDLE variable type 18
- Handlers
  - exception 91–101
  - termination 103–106
- Handles 9, 36
  - duplicating 173
  - inheritable 169–171
  - pseudo 172
- hard link 47
- Hardware Abstraction Layer (HAL) 5
- Heap handle 124, 125
- HEAP\_GENERATE\_EXCEPTIONS flag 96, 124, 125, 126, 128
- HEAP\_NO\_SERIALIZE flag 124, 125, 126, 127, 250
- HEAP\_REALLOC\_IN\_PLACE\_ONLY flag 127
- HEAP\_ZERO\_MEMORY flag 125, 127
- HeapAlloc function 96, 125
  - in Win64 465
- HeapCompact function 128
- HeapCreate function 96, 123
- HeapDestroy function 124
- HeapFree function 126, 153
- HeapLock function 128, 250
- HeapReAlloc function 126
- Heaps 122–129
  - growable and nongrowable 124
  - synchronizing 250
- Heapsize function 127
- HeapUnlock function 128, 250
- HeapValidate function 128
- HeapWalk function 128
- HIGH\_PRIORITY\_CLASS 218

HighPart data type 58  
 HINSTANCE handle 153  
 HKEY\_CLASSES\_ROOT registry key 80  
 HKEY\_CURRENT\_CONFIG registry key 80  
 HKEY\_CURRENT\_USER registry key 80  
 HKEY\_LOCAL\_MACHINE registry key 79  
 HKEY\_USERS registry key 79  
 HMODULE handle 154  
 htonl function 351  
 htons function 351  
 huge files 55

**I**

**I/O**

- alertable 411
- asynchronous 402
- completion ports 422–425
- console 37–39
- extended 411–415
- overlapped 374, 402–406
- standard 35–37, 169

IDLE\_PRIORITY\_CLASS 218  
 ImageName argument 166, 168–169  
 Implicit linking 150–152  
 INADDR\_ANY 350  
 InBuf word 322  
 Index word for keys 82  
 Inherit flag 173  
 Inheritance 173  
 InheritHandles flag 166, 170  
 .INI Windows 3.1 files 79  
 InitializeAcl function 434, 438  
 InitializeCriticalSection  
   function 237, 271  
 InitializeCriticalSectionAndSpin-  
   Count function 238  
 InitializeSecurityDescriptor  
   function 435  
 InitializeSid function 437  
 InitializeUnixSA function 442  
 InitialOwner flag 246  
 InitialState flag 253  
 InitUnFp functions 443  
 In-process servers 363  
 Interfaces 151–152  
 Interlocked functions 234, 260–261  
 InterlockedCompareExchange 261

InterlockedDecrement function 234  
 InterlockedExchange function 260  
 InterlockedExchangeAdd function 261  
 InterlockedIncrement function 234  
 Internet protocol 348  
 Interprocess communication (IPC)  
   one-way 170  
   two-way 319–328  
 INVALID\_HANDLE\_VALUE 122, 322, 423  
 INVALID\_SOCKET 348  
 IP address 350  
 IP storage 24  
 IsValidAcl function 445  
 IsValidSecurityDescriptor  
   function 445  
 IsValidSid function 445  
 Itanium processor 458

**J**

**Job**

- management 186
- objects 194–195

JobMgt  
   displaying active jobs program 192  
   new job information function 190  
   process ID program 193  
 JobMgt.h 483  
 JobShell background job program 187

**K**

Kernel objects 9, 451  
 Key handle 81  
 Key management 81–83  
 KEY\_ALL\_ACCESS flag 81  
 KEY\_ENUMERATE\_SUBKEYS flag 81  
 KEY\_QUERY\_VALUE flag 81  
 KEY\_WRITE flag 81

**L**

LARGE\_INTEGER 183  
   Microsoft C data type 57  
   \_\_leave statement 104  
 LeaveCriticalSection function 237  
 link UNIX function 49  
 Linking  
   explicit 152–154  
   implicit 150–152  
   run-time 152–154

- Linux xxv
  - listen function 350
  - LoadLibrary function 153
  - LoadLibraryEx function 153
  - Local storage 235
  - LocalFileTimeToFileTime function 66
  - LOCKFILE\_EXCLUSIVE\_LOCK flag 74
  - LOCKFILE\_FAIL\_IMMEDIATELY flag 75
  - LockFileEx function 74–75
  - LocsRvr locate the server function 341
  - LONGLONG data type 58
  - LookupAccountName function 436
  - LookupAccountSid function 436
  - LowPart data type 58
  - lpsa pointer 27
  - lpsz prefix 26
  - LPTSTR type 33
  - ls program 69
  - lsFP program 441
  - lsReg program 84
- M**
- MachineName pointer 391
  - MAILSLOT\_WAIT\_FOREVER 338
  - Mailslots 335–339
  - main service entry program 381
  - MakeAbsoluteSD function 453
  - MakeSelfRelativeSD function 453
  - MAKEWORD macro 347
  - malloc C library function 129
  - Managing directories 50–51
  - ManualReset flag 253
  - Map size 137
  - MapName pointer 136
  - MapObject handle 137
  - Mapping 136–140
  - MapViewOfFile function 76, 137
  - MapViewOfFileEx function 137
  - Master-slave scheduling 226
  - MAX\_PATH buffer length 51, 68
  - MAXIMUM\_WAIT\_OBJECTS 176
  - MaximumSize word 123
  - MaximumSizeHigh word 136
  - MaximumSizeLow word 136
  - MaxInstances word 322
  - MaxMsg size 338
  - MCW\_EM mask 98
  - Mem block in heap 127
  - Memory management 119–121
    - performance 262
  - Memory-mapped files 119, 134–140
  - MESSAGE type 355
  - Message waiting 259
  - Microsoft Visual C++ 471
  - mkfifo UNIX function 328
  - mmap UNIX function 138
  - mode UNIX argument 29
  - Mode word 37
  - Models
    - apartment 342
    - boss/worker 211–212
    - client/server 211, 319
    - condition variable 282–287
    - pipeline 211
    - producer/consumer 277, 286
    - threading 211–212, 341–342
    - work crew 211
  - MoveFile function 48–49
  - MOVEFILE\_COPY\_ALLOWED flag 49
  - MOVEFILE\_DELAY\_UNTIL\_REBOOT flag 49
  - MOVEFILE\_REPLACE\_EXISTING flag 48
  - MOVEFILE\_WRITETHROUGH flag 49
  - MoveFileEx function 48–49
  - MoveMethod mode 57
  - Moving
    - directories 48–49
    - files 48–49
  - MSG\_PEEK flag 354
  - MsgWaitForMultipleObjects
    - function 259
  - MsgWaitForMultipleObjectsEx 411
  - MSVCRT\_HEAP\_SELECT 262
  - Multistage pipeline program 299
  - munmap UNIX function 138
  - Mutexes 245–249, 259–260, 282
    - granularity 259
  - MutexName pointer 246
  - Mutual exclusion objects 245–249



**N**

Name string 26  
Named  
    pipes 319–328  
    sockets 350  
Naming  
    conventions 10  
    directories 25, 49  
    drives 25  
    files 25, 49  
New string 48  
NMPWAIT\_NOWAIT named pipe flag 327  
NMPWAIT\_USE\_DEFAULT\_WAIT named  
    pipe flag 327  
NMPWAIT\_WAIT\_FOREVER named pipe  
    flag 327  
nNumberOfLinks 65  
Nongrowable heap size 124  
NORMAL\_PRIORITY\_CLASS 218  
NT  
    services 379–399  
NTFS 24  
NumberOfBytesRead 30  
NumberOfBytesToRead 30

**O**

Objects 176  
    waiting for 259  
offset word 75, 405  
offsetHigh word 75, 137, 405  
offsetLow word 137  
Open systems 7–8  
open UNIX function 29  
OPEN\_ALWAYS flag 27  
OPEN\_EXISTING flag 27  
opendir UNIX function 67  
openFileMapping function 136  
Opening files 26–29  
openMode word 322, 403  
openMutex function 246  
openProcess function 172  
openSCManager function 391  
openSemaphore function 250  
openService function 392  
openThread function 204  
openWaitableTimer function 419

**Operating systems**

    functionality 1–6  
    standards 7–8  
    Win64 11  
options  
    function 43, 486  
    word 124, 174  
    word for keys 81  
outBuf word 322  
Overlapped  
    I/O 374, 402–406  
    sockets 403–404  
overlapped pointer 31, 75  
OVERLAPPED structure 75, 404–406  
OWNER\_SECURITY\_INFORMATION 446  
ownerDefaulted flag 437

**P**

PAGE\_READONLY flag 135  
PAGE\_READWRITE flag 135  
PAGE\_WRITECOPY flag 136  
Paging files 123  
Path directory name 50, 68  
PATH environment variable 169  
Pathnames 25  
PeekNamedPipe function 327  
Peer-to-peer scheduling 226  
Performance 262, 266  
Periodic signal program 420  
Permissions 439–440  
perror C library function 16  
PEXCEPTION\_POINTERS 116  
PF\_INET 348  
pipe program 318  
Pipe size 316  
PIPE\_ACCESS\_DUPLEX flag 322  
PIPE\_ACCESS\_INBOUND flag 322  
PIPE\_ACCESS\_OUTBOUND flag 322  
PIPE\_NOWAIT flag 322  
PIPE\_READMODE\_BYTE flag 322  
PIPE\_READMODE\_MESSAGE flag 322  
PIPE\_TYPE\_BYTE flag 322  
PIPE\_TYPE\_MESSAGE flag 322  
PIPE\_UNLIMITED\_INSTANCES flag 322  
PIPE\_WAIT flag 322  
Pipeline model 211

- Pipes
  - anonymous 316
  - named 319–328
  - summary 338
- Pointer precision data types 462–464
- Pointers
  - based 144
  - file 56–58
- POSIX xxv
- PostQueuedCompletionStatus
  - function 425
- Predefined data types 9
- Prefix string 68
- PrintMsg program 40
- Printstrings function 39
- Priority and scheduling 218–219
- Process
  - components 163–164
  - console 166
  - creation 165–168
  - environment 177
  - identities 172–173
  - inheritance 169–171
  - priority 167
  - priority and scheduling 218–219
  - single 176
  - synchronization 176–177, 236–258
  - waiting for completion 176–177
- Process attribute structure 166
- PROCESS\_ALL\_ACCESS flags 172
- PROCESS\_DUP\_HANDLE flag 173
- PROCESS\_INFORMATION structure 167, 172
- PROCESS\_QUERY\_INFORMATION
  - operation 172
- PROCESS\_TERMINATE operation 172
- ProcessItem function 441
- Producer and consumer program 241
- Producer/consumer model 277, 286
- Protect word 135
- pthread\_cond\_wait UNIX function 284
- Pthreads
  - condition variables 254
  - in POSIX 206
- PulseEvent function 253
- pwd
  - program 51
  - UNIX command 51
- Q**
- qsort C library function 142, 514
- QuadPart data type 58
- QueryJobInformationObject
  - function 195
- QueryServiceStatus function 394
- QueueObj queue management
  - functions 294
- Queues
  - definitions 292
  - in a multistage pipeline 297–304
  - management functions 294
  - object 292–293
- QueueSize connection requests 350
- R**
- Race conditions 235
- RaiseException function 99–101
- read UNIX function 31
- ReadConsole function 38, 41
- readdir UNIX function 67
- ReadFile function 30–31, 316, 322, 403
- ReadFileEx function 412
- ReadFileGather function 31
- ReadFilePermissions program 448
- Reading files 30–31
- ReadTimeout word 338
- realloc C library function 129
- REALTIME\_PRIORITY\_CLASS 218
- ReceiveMessage function 356
- recv function 353
- recvfrom function 373
- ReferencedDomain string 436
- REG\_BINARY registry data type 83
- REG\_CREATED\_NEW\_KEY 83
- REG\_DWORD registry data type 83
- REG\_EXPAND\_SZ registry data type 83
- REG\_OPENED\_EXISTING\_KEY 83
- REG\_OPTION\_NON\_VOLATILE flag 83
- REG\_OPTION\_VOLATILE flag 83
- REG\_SZ registry data type 83
- RegCloseKey function 81
- RegCreateKeyEx function 82

- RegDeleteKey function 83
  - RegDeleteValue function 84
  - REGEDIT32 command 78
  - RegEnumKeyEx function 82
  - RegEnumValue function 81, 83
  - RegEnumValueEx function 84
  - RegisterEventsSource 398
  - Registry 78–81
  - RegOpenKeyEx function 81
  - RegQueryValueEx function 84
  - RegSetValueEx function 81, 84
  - ReleaseMutex function 246
  - ReleaseSemaphore function 250
  - RemoveDirectory function 49
  - RemoveVectoredExceptionHandler 116
  - ReOpenFile function 29
  - ReportError function 42
  - ReportEvent 398
  - ReportException function 102
  - RegQueryValueEx function 81
  - Reserved word 75
  - ResetEvent function 253
  - ResumeThread function 166, 205
  - Revision word 435
  - Run-time linking 152–154
- S**
- SACL\_SECURITY\_INFORMATION 446
  - sbrk UNIX function 125
  - Scheduling 226
  - SCManager handle 392
  - Searching for a file 63–65
  - SEC\_IMAGE flag 135
  - secInfo type 446
  - Secure Sockets Layer (SSL) 454
  - Security
    - identifiers (SIDs) 435–437
    - kernel object 451
    - user object 451
    - Win32 objects 431–454
  - Security attributes 442
    - initialization program 443
  - Security descriptors 432–433, 438–439, 453
    - reading and changing 445–447
  - SECURITY\_ATTRIBUTES structure 170, 432
  - SECURITY\_DESCRIPTOR\_CONTROL
    - flags 435
  - Semaphores 250–253
    - semInitial semaphore value 250
    - semMax semaphore value 250
    - send function 354
    - sendReceivesKHA program 371
    - sendReceivesKST program 367
    - sendto function 373
  - Sequential file processing 14
  - serverCP program 426
  - serverNP program 330
  - Servers
    - in-process 363
    - serverSK program 359
  - Service Control Manager (SCM) 380
  - SERVICE\_AUTO\_START 392
  - SERVICE\_BOOT\_START 392
  - SERVICE\_DEMAND\_START 392
  - SERVICE\_STATUS structure 383–386
  - SERVICE\_STATUS\_HANDLE object 382
  - SERVICE\_SYSTEM\_START 392
  - SERVICE\_TABLE\_ENTRY array 380
  - ServiceMain functions 381–386
  - serviceName pointer 392
  - Services
    - control handler 386–387
    - control handler registration 382
    - control manager 380
    - control program 395
    - controlling 393
    - controls 386
    - creating 391–392
    - debugging 398–399
    - deleting 391–392
    - opening 391
    - setting status 383
    - starting 392
    - state 385
    - status query 394
    - type 384
    - wrapper program 388
  - serviceshell program 395
  - servicespecificExitCode word 384
  - servicestartTable function 380
  - servicestatus function 383
  - servicetype word 384–385, 392

- servstat pointer 393
- setConsoleCtrlHandler function 112
- setConsoleMode function 37–38
- setCriticalSectionSpinCount 238
- setCurrentDirectory function 50
- setEndOfFile function 59
- setEnvironmentVariable function 178
- setEvent function 253
- setFileAttributes function 67
- setFilePointer function 56–57, 405
- setFileSecurity function 446
- setFileShortName 65
- setFileTime function 66
- setInformationJobObject
  - function 194
- setKernelObjectSecurity
  - function 451
- setMailslotInfo function 338
- setNamedPipeHandleState
  - function 323
- setPriorityClass function 219
- setPrivateObjectSecurity
  - function 451
- setProcessorAffinityMask
  - function 221
- setSecurityDescriptorControl
  - flag 435
- setSecurityDescriptorDacl
  - function 439
- setSecurityDescriptorGroup
  - function 437
- setSecurityDescriptorOwner
  - function 433
- setSecurityDescriptorSacl
  - function 453
- setServiceStatus function 383
- setStdHandle function 36
- setThreadIdealProcessor
  - function 221
- setThreadPriority function 219
- setThreadPriorityBoost
  - function 219
- Setting directories 50–51
- setValidFileData 59
- setVolumeMountPoint 68
- setWaitableTimer function 419
- Shared
  - memory in UNIX 139
  - variables 239
- shareMode word 27
- Shortcuts 48
- shutdown function 351
- SID management 454
- SID\_NAME\_USE variable 436
- sidTypeGroup value 436
- sidTypeUser value 436
- sidTypeWellKnownGroup value 436
- Signaled state 206
- Signaling producer and consumer
  - program 255
- signalObjectAndWait function 285, 411
- Signals 113, 167
  - in UNIX 102
- simplePC program 241
- simpleService program 388
- 64-bit
  - addressing 459
  - architecture 459
  - file addresses 55–56
  - migration 460
  - UNIX systems 460
- size for heap allocation 126
- SIZE\_T data type 465
- skipArg function 183, 487
- sleep function 223
- sleepEx function 413
- SOCK\_DGRAM 373
- sockaddr structure 349
- sockaddr\_in structure 349
- socket function 348, 404
- SOCKET\_ERROR 349, 353, 354
- Socket-based
  - client program 357
  - server program 359
- Sockets
  - Berkeley 346, 374
  - binding 348–350
  - client functions 352–355, 356
  - closing 351
  - connecting to client 350–351
  - connecting to server 352–353
  - creating 347–348
  - disconnecting 351

- initialization 346–347
- message receive 355–356
- named 350
- overlapped 403–404
- server functions 348–352, 358
- SOMAXCON 350
- sort UNIX command 129
- sortBT
  - performance 514
  - program 131
- sortFL
  - performance 514
  - program 142
- sortFLSR program 514
- sortHP performance 514
- sortMM
  - based pointers program 145
  - creating the index program 148
  - performance 514
- sortMT
  - merge-sort program 213
- source handle 173
- Sparse file 59
- Spin 238, 261
- svrBcst mailslot client program 340
- SSIZE\_T data type 123
- stack byte size 203
- Stack unwind 105–106
- Standard
  - I/O 35–37, 169
  - input 170
- startAddr function pointer 203
- STARTF\_USESTDHANDLES flags 167
- startInfo string 167
- startService function 393
- startServiceCtrlDispatcher
  - function 380
- startType word 392
- stat UNIX function 67
- STATE\_TYPE data structure 283
- statsMX program 266
- Status functions for named pipes 323
- STATUS\_ACCESS\_VIOLATION exception code 128
- STATUS\_NO\_MEMORY exception code 96, 128
- STD\_ERROR\_HANDLE flag 36
- STD\_INPUT\_HANDLE flag 36
- STD\_OUTPUT\_HANDLE flag 36
- stdErr handle field 167
- stdHandle word 36
- stdIn handle field 167
- stdout handle field 167
- STILL\_ACTIVE process status 175, 204
- Storage 235
- Storage area networks (SANs) 24
- Strings, environment 177–178
- Structured Exception Handling (SEH)
  - 91–92, 106
- Structures, overlapped 404–406
- subKey handle 81
- support.h 481
- suspendThread function 205
- Symmetric multiprocessing (SMP) 163
- synchObj
  - queue definitions 292
  - threshold barrier definitions 289
- Synchronization 218–219
  - heap 250
  - objects 411
  - performance impact 266
  - processes 176–177
  - processes and threads 236–258
- SYNCHRONIZE
  - flag 172
  - named pipes 450
- System
  - ACLs (SACLs) 433, 453
  - error codes 18
  - include files 10
  - system pointer 436
  - systemTimeToFileTime function 66
- T**
  - \_t\_access function 44, 441
  - tail program 61
  - tchar.h 33
  - TCHAR type 32
  - TCP/IP 346, 348
  - Telnet 352
  - TempFile pointer 68
  - TemplateFile handle 28
  - Temporary file names 67–68

- TerminateProcess function 172, 175, 186
  - TerminateThread function 204, 206
  - Termination handlers 103–106
  - Testing 309–311
  - testTHB test program 290
  - THB\_HANDLE threshold barrier handle 289
  - ThbObject threshold barrier
    - implementation 290
  - Thread attribute structure 166
  - Thread Local Storage (TLS) 164, 202, 216–217
  - THREAD\_PRIORITY\_ABOVE\_NORMAL 219
  - THREAD\_PRIORITY\_BELOW\_NORMAL 219
  - THREAD\_PRIORITY\_HIGHEST 219
  - THREAD\_PRIORITY\_IDLE 219
  - THREAD\_PRIORITY\_LOWEST 219
  - THREAD\_PRIORITY\_NORMAL 219
  - THREAD\_PRIORITY\_TIME\_CRITICAL 219
  - ThreadParm thread argument 203
  - Threads
    - common mistakes 222–223
    - creating 202–203
    - file locking 74–78
    - identity 204
    - models 211–212, 341–342
    - overview 199–200
    - primary 165
    - priority and scheduling 218–219
    - remote 206
    - resuming 205
    - single 163–164
    - states 220–222
    - statistics program 266
    - storage 202, 216–217
    - suspending 205
    - synchronization 218–219, 236–258
    - terminating 204
    - waiting for termination 205–206
    - with asynchronous I/O 418
    - with the C library 207–208
  - Thread-safe
    - code 231–236
    - DLL program 367
    - DLL program with state structure 371
    - libraries 208
  - Threestage multistage pipeline
    - program 299
  - Threshold barrier object 289–292
  - time UNIX command 183
  - TimeBeep program 420
  - Timed waits 223
  - Timeout word 176, 322, 327
  - timep
    - performance 505
    - process times program 183
  - Timers
    - waitable 419–420
  - TLS\_MINIMUM\_AVAILABLE 217
  - TlsAlloc function 217
  - TlsFree function 217
  - TlsGetValue function 218
  - TlsSetValue function 218
  - \_tmain function 34, 169
  - touch program 72
  - Toupper program 107
  - TransactNamedPipe function 326, 403
  - Transfer pointer 406
  - TRUNCATE\_EXISTING flag 27
  - \_\_try 92
  - TryEnterCriticalSection
    - function 238
  - Try-except blocks 92–93, 103–105
  - Try-finally blocks 103–105
  - Type for registry 83
- U**
- ULONGLONG data type 58
  - Unicode 32–34
  - Unicode UTF-16 32
  - unique suffix value 68
  - Universal Coordinated Time (UCT) 66
  - Universal Disk Format (UDF) 24
  - unlink UNIX function 49
  - unlockFileEx function 75
  - UnmapViewOfFile function 137
  - Unwinding stacks 105
  - utime UNIX function 67
- V**
- va\_arg C library function 39
  - va\_end C library function 39
  - va\_start C library function 39
  - Value management 83–84
  - Variables, environment 177–178

- VectoredHandler 116
- VersionRequired word 347
- Virtual
  - address space 120
  - memory manager 121
  - memory space allocation 136
- Visual C++ 10, 21
- volatile
  - storage modifier 310
  - variables 236, 239
- W**
- Wait
  - for messages and objects 259
  - functions 413–415
- wait flag 406
- WAIT\_ABANDONED 247
- WAIT\_ABANDONED\_0 177, 247
- WAIT\_FAILED 177
- WAIT\_OBJECT\_0 return value 177
- WAIT\_TIMEOUT 177
- Waitable timers 419–420
- waitAll parameter 177
- waitForMultipleObjects function
  - 176–177, 205, 245, 254
- waitForMultipleObjectsEx
  - function 413
- waitForSingleObject function
  - 176–177, 205, 245
- waitForSingleObjectEx function 413
- waitHint word 384
- Waiting for a process 176–177
- waitNamedPipe function 324
- waitThresholdBarrier function 291
- wchar.h 34
- WCHAR type 32
- Win16 compatibility 10–11
- Win32
  - API 2
  - principles 8–11
- win32ExitCode word 384
- Win64
  - API 458
  - code migration 464–466
  - data types 462–463
  - introduction 11
  - programming 457–466
- Window appearance 167
- Windows
  - Sockets 346, 374
  - Sockets 2 375
- Windows 2003 Server 29
- windows.h file 17
- Winsock 454
  - API 346
  - Initialization 346–347
- Work crew model 211
- write UNIX function 31
- WRITE\_DAC permission 446
- writeConsole function 38
- writeFile function 31, 403
- writeFileEx function 412
- writeFileGather function 31
- Writing files 31
- WS2\_32.DLL 347
- WSA\_FLAG\_OVERLAPPED flag 403
- WSACleanup function 347
- WSADATA pointer 347
- WSADATA structure 347
- WSAGetLastError 347
- WSASocket function 403
- WSAStartup function 347