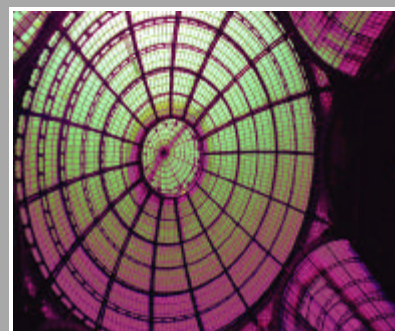
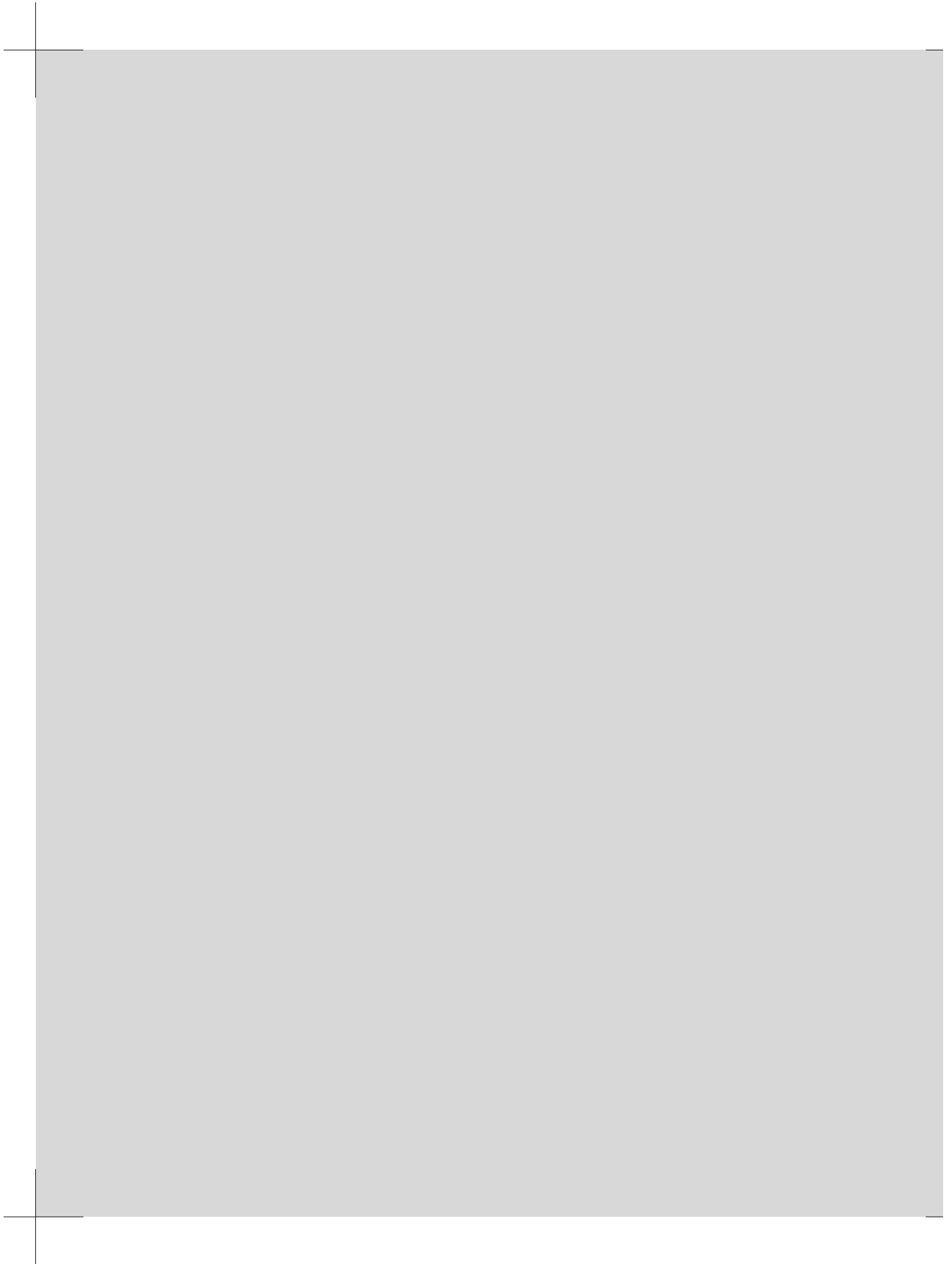


Win 32

Win 64

1





Microsoft Windows (operating system, OS)
 Win32 API(application programming interface,
) . 64 Win64 API , Win32
 Win64 . , Win32 Win64 Windows
 Windows API . Win32 Win64
 가 . Windows ,
 API ,
 가 .
 OS API 가 , Windows API
 . Windows . Windows
 ,
 ,
 C
 Windows .
 Windows , , PC,
 OS OS가
 OS가 . OS가
 . OS (flat) 가 , ,
 . OS ,
 OS 가 .
 (naming) ,
 . OS
 . OS
 (preempted)[†] ,

† : (preempted) 가

. OS

. OS

Microsoft Windows Win32/Win64 API OS
Windows
API

Windows

Windows API Windows

Windows Windows API

가

Windows API 가 , API
. Windows :

가 (scalability).
가)

(

가

가

API

API

가

. API

Windows

1993 Windows

Microsoft

Home, Professional

Windows XP
PC, ,

Windows XP

Windows Server 2003 , Small Business
 Server, Storage Server 2003 . Windows 2003(“Server”
)
 (symmetric multi processing) 가 . Win64 64
 Windows 2003
 Professional Server Windows 2000
 Windows 2000 , Windows XP
 Windows
 Windows Embedded, Windows CE, Tablet PC, Windows Mobile
 Windows , Windows

Windows

Microsoft가

Windows

1993 Windows NT 3.5, 3.5.1, 4.0 NT . 가
 Windows NT Version 4.0 Service Pack 3(SP 3) .
 Windows 9x , NT 가
 . Windows 2000 NT Version 5.0 ,
 Windows 2000 , 2003, XP Windows NT NT5 .
 NT Version 4.0 (.
 Windows 95, Windows 98, Windows ME(가
 Windows 9x) ,
 가 NT Windows XP
 , Windows XP NT
 , Windows 9x
 가 , 16 OS Windows 3.1 . Windows 3.1
 Windows 95가 . Windows 3.1
 (GUI) Windows GUI . Windows 3.1 API
 , , OS .

1980 DOS “IBM PC” OS가 . DOS
 , DOS
 DOS (batch file)

Windows NT 5.x

Windows 2000 XP, 2003 NT Version 5.x NT5
 Windows NT 5 , (“5.x” “x”)
 Windows XP NT 5.1
 Windows NT5 가
 Windows NT5 가
 NT Windows 9x
 Windows

Microsoft API NT, Windows(9x), CE
 Windows API 가

Windows

Windows(Win32, Win64 API, NT5) “
 ” 가
 . NT OS
 Windows 가 가 . Microsoft가
 (Resource Kit) POSIX , POSIX

Windows 가 ,
 가 (Hardware Abstraction Layer,
 HAL) . HAL 가 .
 Windows Intel x86 . x86
 Pentium Xeon , 486 . Advanced Micro
 Devices(AMD) . Windows
 . 가 , x86 64
 Intel Itanium Windows 2003 .
 Windows 가 :
 Windows CE x86 .
 Windows NT Digital Equipment Corporation (Compaq HP가) Alpha
 가 .
 AMD Athlon 64 Opteron 64 (AMD64) x86 64
 . Itanium
 Intel 32/64 64 x86 .

Windows

OS 가 Windows .
 OS 가 , UNIX¹⁾ Linux
 Windows , Windows
 .
 Windows ,
 2) Windows 가 UNIX Linux, Macintosh 가
 가 .

1) UNIX Linux POSIX API .

2) Windows Linux가 가 .
 Linux Windows

Windows , Windows
 가 Windows
 Windows GUI
 Windows
 “ ”
 Windows SMP . Windows . Windows
 3)
 Windows National Security Agency(NSA) C2 (Windows 9x C
).
 UNIX Linux, Windows OS
 Windows OS UNIX (, UNIX).
 C2 NT
 Windows OS ,
 가 , Windows
 Windows

Windows ,

Windows API . UNIX †
 “Windows 가?”, “Windows 가?”,
 “Windows API 가?” 가
 ,

3) RAM 16MB 486 가 4 RAM 8GB 2GHz Xeon , Windows

† : open system. Linux POSIX

Windows API Linux UNIX가 POSIX API . Windows X/Open ,

Windows 가 . Microsoft가 , Windows Windows 가

Windows 가 OS 가 가

가 Windows 가

Windows , Windows C, C++ . Windows

(Sockets) TCP/IP , Windows (RPC) 가 ⁴⁾ SQL(Structured Query Language) (DBMS)

Windows . Windows TCP/IP , XWindows Windows 가

, Windows , API 가

4) Windows RPC Windows 가

가). , 가 가 가
 , 가
 Windows
 Windows
 Windows Windows
 가
 Wait For Single Object
 Wait For Single Object Ex
 Wait For Multiple Objects
 Wait Named Pipe
 , (data type) 가
 API
 :
 BOOL (, 32)
 HANDLE
 DWORD (32)
 LPTSTR (8 16)
 LPSECURITY_ATTRIBUTES
 * , LPTSTR(TCHAR *) LPCTSTR(const
 TCHAR *) const : TCHAR
 char 2 wchar_t
 lpzFileName "0 (zero-terminated string, null
) 가 (long pointer) dwAccess
 (32) . dw가 (double word)
 가
 가
 : , , ,
 . Microsoft Visual C++
 Microsoft Visual Studio .NET\vc7\PlatformSDK\Include Program Files\
 Program Files\Microsoft Visual Studio\VC98\Include (VC++ 6.0

WINDOWS.H ()
 WINNT.H
 WINBASE.H
 , Win32 API , Windows 3.1
 Win 16 API
 가 .
 LPSTR LPDWORD LP long
 pointer ()가 32 64 .
 “long” LPVOID PVOID
 6)
 WIN32_FIND_DATA “WIN32”가 Win64
 16 Windows , 16
 OpenFile
 , 16
 CreateFile .

Win64

Win64 AMD AMD64(Opteron Athlon 64) Intel
 Itanium (Merced, McKinley, Madison, IA-64
) Windows XP 2003 . Win64
 . Win32 Win64 가
 (Win64 64) 가 .
 Win64 . ,
 Win64
 64
 , ()

6) PVOID L L Microsoft

(LONG, DWORD) 가 가
 DWORD32 DWORD64가
 POINTER_32 POINTER_64가
 , Win32 Win64
 Windows Win32 API . 16
 Win64

UNIX Linux Windows
 , Windows HANDLE “ ” . HANDLE
 . UNIX 0, 1, 2가
 Windows
 ID Windows . Windows
 HANDLE , , , ,
 Windows 가 . Windows UNIX
 . VMS OpenVMS HP(DEC Compaq)
 VMS David Cutler가 NT Windows
 가 . , Windows (job
 object)
 , UNIX LF Windows
 CR-LF .

C :
 Windows , (2 3
) C ANSI C
 . C (ANSI)
 <string.h>, <stdlib.h>, <signal.h>, I/O , I/O Windows

가
<stdio.h> fopen fread
C , Windows
? C++ I/O NET I/O
가 , Windows
Windows C C++ I/O
가 , Windows가
C Windows
3
. C Windows
(), I/O, (가 4GB)
C . C
Windows
5 C
C
가
Windows
Microsoft Visual Studio .NET Microsoft Visual C++ Version 6.0 C
. Microsoft 가
A 가
: Windows , Microsoft Visual Studio
.NET

RAM

가 7)

CD-ROM ()

Microsoft Visual C++

가 Microsoft

:

가

1. C
2. Windows
3. Windows CopyFile

C Windows

가 , 가

Windows

7) 1997

16MB,

256MB

1000
(RAM PC

),

100 ,

2500 PC
50

RAM 10

, I/O ,
 , 가
 Windows , UNIX
 cp
 ,
 Windows
 : UNIX
 C
 1-1 , C FILE I/O
 FILE 1-2 Windows HANDLE ,

1-1 cpC: C

```

/* 1 . cp
   C . */
/* cpC file1 file2: file1 file2 . */
#include <stdio.h>
#include <errno.h>
#define BUF_SIZE 256

int main (int argc, char *argv [])
{
    FILE *in_file, *out_file;
    char rec [BUF_SIZE];
    size_t bytes_in, bytes_out;
    if (argc != 3) {
        printf ("Usage: cpC file1 file2\n");
        return 1;
    }
    in_file = fopen (argv [1], "rb");
    if (in_file == NULL) {

```

```

    perror (argv [1]);
    return 2;
}
out_file = fopen (argv [2], "wb");
if (out_file == NULL) {
    perror (argv [2]);
    return 3;
}
/* . */
while ((bytes_in = fread (rec, 1, BUF_SIZE, in_file)) > 0) {
    bytes_out = fwrite (rec, 1, bytes_in, out_file);
    if (bytes_out != bytes_in) {
        perror ("Fatal write error.");
        return 4;
    }
}
fclose (in_file);
fclose (out_file);
return 0;
}

```

-
- 가 가
- . , Windows .
1. FILE 가 (UNIX
). NULL 가 .
 2. fopen . (end-of-line, EOL) 가
. Windows ,
I/O (C
) .
 3. perror . perror errno 가
FILE
perror ferrror .

4. fread fwrite 가

가 , 0

5. fclose FILE (UNIX)

6. I/O , I/O 가

7. C I/O printf Windows

C UNIX Windows, ANSI C C

I/O 가 , 7 Windows

UNIX 가 , C

(fseek fsetpos

fgetpos). 가 C I/O

: Visual C++ , C

: I/O I/O C

Windows

Windows

1-2 Windows API ,

1-2 cpW Windows ,

```

/* 1 . cp . Windows . */
/* : cpW 1 2 . */
1 2 . */
    
```

```
#include <windows.h>
#include <stdio.h>
#define BUF_SIZE 256
int main (int argc, LPTSTR argv [])
{
    HANDLE hIn, hOut;
    DWORD nIn, nOut;
    CHAR Buffer [BUF_SIZE];
    if (argc != 3) {
        printf ("Usage: cpWfile1 file2\n");
        return 1;
    }
    hIn = CreateFile (argv [1], GENERIC_READ, 0, NULL,
        OPEN_EXISTING, 0, NULL);
    if (hIn == INVALID_HANDLE_VALUE) {
        printf ("Cannot open input file. Error: %x\n",
            GetLastError ());
        return 2;
    }
    hOut = CreateFile (argv [2], GENERIC_WRITE, 0, NULL,
        CREATE_ALWAYS, FILE_ATTRIBUTE_NORMAL, NULL);
    if (hOut == INVALID_HANDLE_VALUE) {
        printf ("Cannot open output file. Error: %x\n",
            GetLastError ());
        return 3;
    }
    while (ReadFile (hIn, Buffer, BUF_SIZE, &nIn, NULL) &&nIn > 0) {
        WriteFile (hOut, Buffer, nIn, &nOut, NULL);
        if (nIn != nOut) {
            printf ("Fatal write error: %x\n", GetLastError ());
            return 4;
        }
    }
    CloseHandle (hIn);
    CloseHandle (hOut);
    return 0;
}
```

1. <windows.h> , Windows
8)
2. Windows HANDLE
CloseHandle .
3.
가 OS가 ,
(:)
4. Windows
가 INVALID_HANDLE_VALUE
GENERIC_READ .
5. ReadFile WriteFile 가
9) 가 0 . 가 0
6. DWORD GetLastError . Windows
가 2-2 .
7. Windows NT . 15
8. CreateFile ,

Windows

Windows

(C).

CopyFile ,

8) A

9) ANSI C가 “and”(&&) “or”(||) 가

가 (.
256). CopyFile .

1-3 cpCF: Windows

```

/* 1 . cp . ,
   CopyFile Windows . */
/* cpCF file1 file2: file1 file2 . */
#include <windows.h>
#include <stdio.h>
int main (int argc, LPTSTR argv [])
{
    if (argc != 3) {
        printf ("Usage: cpCF file1 file2\n");
        return 1;
    }
    if (!CopyFile (argv [1], argv [2], FALSE)) {
        printf ("CopyFile Error: %x\n", GetLastError ());
        return 2;
    }
    return 0;
}

```

Windows C
가 . C
Windows Windows
, Windows
Windows 가
Windows NT5(XP, 2000, 2003)
, NT Windows 9x(95, 98, Me)

2 3 I/O . I/O, ASCII

Win32

Win32

Marshall Brain Ron Reeves *Win32 System Services* Jeffrey Richter *Programming Applications for Microsoft Windows(Advanced Windows)* 가 .
Windows 95 Windows NT

Microsoft Visual C++

Microsoft <http://www.microsoft.com>

Windows . MSDN(Microsoft Developer's Network) 가

Win64

Win64 , Microsoft

16

Windows NT Windows NT

Windows Windows

David Solomon Mark Russinovich *Inside Windows 2000* .

, 가 , , I/O , API

Windows 9x, CE

- Solomon, NT Helen Custer
가 .
- UNIX
(故) W. Richard Stevens *Advanced Programming in the UNIX Environment*
Windows UNIX
Stevens UNIX ,
. UNIX 가 , UNIX(Linux)가
C I/O UNIX I/O , Windows
가 .
- OS UNIX 가 Eric S. Raymond *The Art of UNIX Programming* . , Windows
- Windows GUI
Windows Brent Rector
Joseph M. Newcomer *Win32 Programming* Charles Petzold *Programming Windows, Fifth Edition* .
- OS Abraham Silberschatz
*Operating System Concepts*가 가 .
- ANSI C
P. J. Plauger *The Standard C Library* 가 .
, Brian W. Kernighan Dennis M. Ritchie *The C Programming Language*가 . C . C

Windows CE

Windows CE Jason P. Nottingham, Steven Makofsky, Andrew Tucker *SAMS Teach Yourself Windows CE Programming in 24 Hours*

UNIX Windows

Unix X Windows API Wine
가 <http://www.winehq.com>



1-1. 가 , , . Microsoft Visual C++
, UNIX (Microsoft Visual C++). :
. A Microsoft Visual Studio
.NET Visual C++ 6.0

1-2. Microsoft Visual Studio .NET Visual C++
. A , Microsoft
가 .

1-3. Windows (CR-LF)
1-1 ? UNIX ?

1-4.
, :
가 . 6
, C 가 가 .