CSC415 – Homework #1

Aleksandr Kibis

9/1/2014

The goal of this assignment was to get an introduction to high level system calls. I wrote two programs, in C, which wrote a string to console. One for Windows (Win32) and the other for Linux (POSIX). The two programs were very similar but still required a bit of research to really understand how to put them together. There were three main functions which were used: sprintf(Linux/Windows), write(POSIX), and WriteFile(Win32). Sprintf is quite similar to printf, except instead of writing directly to console, a string is written to a buffer provided to it (a pointer to a char array in my case). This is about the point where the two programs seized to be similar as a specific function was used for each environment. The POSIX version was much simpler to write but did not provide documentation that was as detailed as the Win32 implementation. I ran into a problem with file handlers/descriptors since there was no file being opened or closed. The solution was different between the two environments. For windows, a new "HANDLE" variable had to be created which pointed to stdout. For Linux, all I had to do was set the handler/descriptor parameter to "1" in order to achieve the same thing.

Windows

Code

```
/**
File: main.c
Author: Aleksandr Kibis
Date: 9/1/2014

This program uses a high level system call to write a string to console on a
Windows-based system.
**/

#include <windows.h>
#include <stdio.h>


#define NAME "Aleksandr Kibis"
int main(){
        char buf[50]; // create array to hold string
        int n; // number of bytes written to buffer
        DWORD bytesWritten; // number of bytes written by WriteFile function
        HANDLE stdOut = GetStdHandle(STD_OUTPUT_HANDLE); // get standard output handle

        n = sprintf(buf, "Hello %s, welcome to 415!\n", NAME);

        WriteFile(stdOut, buf, n, &bytesWritten, NULL);

        return 0;
}
```

Output

```
C:\Users\rusky\Dropbox\Fall 2014\650\Homework 1\Submission\Windows>cl -o test main.c
```

```
Microsoft (R) C/C++ Optimizing Compiler Version 17.00.61030 for x64

Copyright (C) Microsoft Corporation.  All rights reserved.


cl : Command line warning D9035 : option 'o' has been deprecated and will be removed in a future release

main.c

Microsoft (R) Incremental Linker Version 11.00.61030.0

Copyright (C) Microsoft Corporation.  All rights reserved.


/out:main.exe

/out:test.exe

main.obj


C:\Users\rusky\Dropbox\Fall 2014\650\Homework 1\Submission\Windows>test.exe

Hello Aleksandr Kibis, welcome to 415!
```

Linux

Code

```c
/*
 * File:   main.c
 * Author: netdom
 *
 * Created on August 29, 2014, 5:49 AM
 */

#include <unistd.h>
#include <stdio.h>

#define NAME "Aleksandr Kibis"
int main() {
    char buf[50]; // create array to hold string
    int n;  // number of bytes written to buffer and by write function
    ssize_t check; // error check

    n = sprintf(buf, "Hello %s, welcome to 415!", NAME);

 //   printf("%d\n", n);

    check = write(1,buf,n);

    return (0);
}
```

Output

```
netdom@kali:~/Desktop/650/Linux$ gcc -o test main.c

netdom@kali:~/Desktop/650/Linux$ ./test
```

```
Hello Aleksandr Kibis, welcome to 415!

netdom@kali:~/Desktop/650/Linux$
```

Virtual Machine Screenshot

Host: Windows 8.1 x64 Professional

Guest: Kali Linux x64 1.0.8

Software: VMWare Player