Programming Project 3

CIS355 – Spring Term 2018

Point Value: 100 points

Project Due Date: **Thursday, April 26, 2018**

Submission Instruction

Please submit your source code file to Schoology. The name of the file should be P3_YourLastname_YourFirstnme.cpp. Please submit the C++ source code file on Schoology by 11:59pm on the due day.

Program Description

Write a (POSIX) program that searches for a specified file, and if found, prints out the relative path of the file from the current working directory and its time stamp. Your program should take one command line argument, the filename to find.

If your executable file is “searchFile” and you run this program from the root directory (“/”) and search for a file named “test.txt”. Your output might be something like the following:

```
$./searchFile test.txt

Searching for test.txt …

Found test.txt in /home/gtian/CIS355/

Last Edited: Nov 4 11:20
```

To start searching inside the current working directory you need to obtain a pointer to a *DIR* structure using the *opendir()* system call. The *DIR* structure provides you a method of getting a pointer to a *dirent* structure (directory entry) for each file or directory within the directory represented by *DIR*. You access each *diren*t structure by repeatedly calling the *readdir()* system call. Each time you call this with a given *DIR* * the function returns a pointer to a different *dirent* structure until there are no more within the directory, in which case *readdir()* return NULL. Within the *dirent* structure is information concerning its type (regular file or directory), its name, size, and timestamp information. For example if I wanted to print out the name of every file or directory within /home/gtian/, the following code would do the trick:

```
DIR * directory;
directory = opendir(“/home/gtian”);
struct dirent * dptr;
while ( dptr = readdir(directory)){
        cout << dptr->d_name << endl;
}
```

Of course, you will need to do a recursive search, in which case you will need to be able to determine if a given *dirent* structure is a directory or a file. To do this, you need to use the *stat()* system call, which takes a *stat* structure as a parameter along with the name of the file and fills in the *stat* structure. To determine if the file is a directory, use the *stat.st_mode* field and a bit mask as shown below:

```
struct stat statStruct;
stat(dptr->d_name, &statStruct);
if( (statStruct.st_mode & S_IFMT) == S_IFDIR){
        cout << “This is a directory” << endl;
}
```

When you are doing your recursive search through directories, do not traverse the “.” And “..” entries. They are the current directory (.) and the parent directory (..). You do not want to go into those two directories.

In order to print out the timestamp, you may use the following function:

```
void getTimestamp(struct dirent *dptr , char * dateString, int strLen) {

      struct tm  *tm;

      struct stat statStruct;
      stat(dptr->d_name, &statStruct )

      tm = localtime(&statStruct.st_mtime);

      /* Get localized date string. */
      strftime(datestring, sizeof(datestring), "%b %d %R", tm);
}
```

To summary, you will need to research the following system calls and data structures in order to complete this project. These system calls and data structures are ONLY AVAILABLE in the UNIX/Linux environment. Do NOT use Windows

System Calls:

```
DIR * opendir(char * pathname);
struct dirent *readdir(DIR *dirp);
int stat(char * d_name, struct stat * statStruct );
void chdir(char * pathname) * // to change current directory
void closedir(DIR *dirp) // to close the DIR structure
```

- depending on how you implement your program, you might not need chdir()

Data Structures:

```
DIR
dirent
stat
```

In order to use these POSIX calls and structures you will likely need to include the following:

```
#include <sys/types.h>
#include <sys/stat.h>
#include <dirent.h>
#include <unistd.h>
#include <time.h>
#include <stdint.h>
#include <locale.h>
#include <langinfo.h>
#include <fcntl.h>
```

Note: Your program must be compiled and executed on the Ubuntu server, **kermit.fontbonne.edu,** or other Linux/Unix system.

Steps to log into **mathcswc.fontbonne.edu:**

1. Open SSH client on your computer.
2. Enter the hostname, username, and port number (22) in the SSH client window just like the figure.
3. You will be prompted to enter your password. Please enter your password and enter.
4. You should log in to the server successfully.