

I expected something like this

Windows:

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

int is_dir(const char *szPath) {
    DWORD dwAttrib = GetFileAttributesA(szPath);
    return (dwAttrib != INVALID_FILE_ATTRIBUTES &&
            (dwAttrib & FILE_ATTRIBUTE_DIRECTORY));
}

void test(const char* name) {
    printf("is_dir=%d : %s\n",is_dir(name),name);
}

int main(int argc,char** argv) {
    test("C:\\Windows");
    test("C:\\Windows\\");
    test("C:\\Windows\\.");
    test("C:\\Windows\\.\\");
    test("C:\\Windows\\none");
    test("\\\\?\\C:\\Windows");
    test("\\\\?\\C:\\Windows\\none");
    return 0;
}
```

Output:

```
is_dir=1 : C:\Windows
is_dir=1 : C:\Windows\
is_dir=1 : C:\Windows\.
is_dir=1 : C:\Windows\.
is_dir=0 : C:\Windows\none
is_dir=1 : \\?\C:\Windows
is_dir=0 : \\?\C:\Windows\none
```

Linux:

```
include <sys/stat.h>
#include <stdio.h>

int is_dir(const char* name) {
    struct stat s;
```

```

    int err = stat(name, &s);
    if(-1 == err) return 0;
    return S_ISDIR(s.st_mode) ? 1:0;
}
void test(const char* name) {
    printf("is_dir=%d : %s\n",is_dir(name),name);
}
int main(int argc,char** argv) {
    test("/home");
    test("/home/");
    test("/home/.");
    test("/home/.");
    test("/home/./");
    test("/home/none");
    return 0;
}

```

Output:

```

is_dir=1 : /home
is_dir=1 : /home/
is_dir=1 : /home/.
is_dir=1 : /home/./
is_dir=0 : /home/none

```

https://en.wikipedia.org/wiki/Principle_of_least_astonishment
