

Operating Systems Programming Assignment 1

By Luke Hanks hanks1@unm.edu

Part 1 STRACE a small program

Create your own program in C as follows:

```
#include <stdio.h>

int main(void)
{
    FILE *fd;
    float pval = 3.14159;
    int i, k;
    if ((fd = fopen("myTstFile", "r+")) == NULL)
        printf("\n Program Failed, figure out why...\n");
    else
    {
        printf("\n Simple pie value %1.8f\n", pval);
        for (i = 0; i < 100; i++)
        {
            k = rand() % 10;
            if (fprintf(fd, "%f\n", pval + i * k) == -1)
                perror("write err");
            fflush(fd);
            printf(".");
            fflush(stdout);
        }
        fclose(fd);
        printf("\n Program successful ends\n");
    }
}
```

a. Without creating file *myTstFile* (no such a file in your working directory), run “strace yourCode”, read your output, pick the first five of most frequently invoked system calls and explain which among the five system calls caused the program to fail.

Most called system calls:

1. open
2. mmap
3. mprotect
4. write
5. munmap

I believe that open caused the program to fail because *myTstFile* isn't in the working directory.

b. With file myTstFile (a dummy file) being created, run "strace yourCode" again, read your output to pick the most frequently invoked system call.

Most frequently invoked system call is write.

c. Is "fopen" a system call? If not, which system call it mainly correlates with?

fopen is not a system call. It is most correlated with the open system call.

d. Is "printf" a system call? If not, which system call it mainly correlates with?

fprint is not a system call. It is most correlated with the write system call.

Part 2 STRACE a Linux utility command -- cal

Run "strace -c cal", capture the output, and then pick the top three system calls which consumed the system time and briefly describe their functionality.

1. write is used to write to a file or to the terminal.
2. mmap is used to map some number of bytes into memory.
3. open is used to open files.

Part 3 STRACE/LTRACE Linux utility commands "ls"

Command ltrace is another tracing tool used for tracing the library function calls. Use both strace and ltrace to Linux command ls to report what library functions and system calls are used to:

a. Open the current directory

- Library call: opendir
- System call: open

b. Get the list of directory entries

- Library call: readdir
- System call: read

c. Print the output to your screen

- Library call: strcoll
- System call: write

By Luke Hanks hanksl@unm.edu