Operating System
Blen Daniel Assefa

OS 2021 Problem Sheet #1

Problem 1.1: library and system calls

a)

$ ltrace /bin/date
```
strrchr("/bin/date", '/')                       = "/date"
setlocale(LC_ALL, "")                           = "en_US.UTF-8"
bindtextdomain("coreutils", "/usr/share/locale") = "/usr/share/locale"
textdomain("coreutils")                         = "coreutils"
__cxa_atexit(0x556ecb5c79b0, 0, 0x556ecb5df248, 0x736c6974756572) = 0
getopt_long(1, 0x7ffff55e0a68, "d:f:I::r:Rs:u", 0x556ecb5de260, nil) = -1
nl_langinfo(0x2006c, 0, 0, 0)                   = 0x7f0eb9f17135
getenv("TZ")                                    = nil
malloc(128)                                     = 0x556ecb9e9440
clock_gettime(0, 0x7ffff55e0890, 1, 0x556ecb9e94c0) = 0
getenv("TZ")                                    = nil
localtime_r(0x7ffff55e07c0, 0x7ffff55e07d0, 0x556ecb5d7128, 8) = 0x7ffff55e07d0
strcmp("", "EST")                               = -69
strlen("EST")                                   = 3
memcpy(0x556ecb9e9449, "EST\0", 4)              = 0x556ecb9e9449
strftime(" Sun", 1024, " %a", 0x7ffff55e07d0)   = 4
fwrite("Sun", 3, 1, 0x7f0eba0d76a0)             = 1
fputc(' ', 0x7f0eba0d76a0)                      = 32
fputc('0', 0x7f0eba0d76a0)                      = 48
fwrite("9", 1, 1, 0x7f0eba0d76a0)               = 1
fputc(' ', 0x7f0eba0d76a0)                      = 32
strftime(" Jan", 1024, " %b", 0x7ffff55e07d0)   = 4
fwrite("Jan", 3, 1, 0x7f0eba0d76a0)             = 1
fputc(' ', 0x7f0eba0d76a0)                      = 32
fwrite("2022", 4, 1, 0x7f0eba0d76a0)            = 1
fputc(' ', 0x7f0eba0d76a0)                      = 32
strftime(" 04:32:15 AM", 1024, " %r", 0x7ffff55e07d0) = 12
fwrite("04:32:15 AM", 11, 1, 0x7f0eba0d76a0)    = 1
fputc(' ', 0x7f0eba0d76a0)                      = 32
strlen("EST")                                   = 3
fwrite("EST", 3, 1, 0x7f0eba0d76a0)             = 1
__overflow(0x7f0eba0d76a0, 10, 3, 0x5453Sun 09 Jan 2022 04:32:15 AM EST
)         = 10
__fpending(0x7f0eba0d76a0, 0, 0x556ecb5c79b0, 1) = 0
fileno(0x7f0eba0d76a0)                          = 1
__freading(0x7f0eba0d76a0, 0, 0x556ecb5c79b0, 1) = 0
__freading(0x7f0eba0d76a0, 0, 0x556ecb5c79b0, 1) = 0
fflush(0x7f0eba0d76a0)                          = 0
fclose(0x7f0eba0d76a0)                          = 0
__fpending(0x7f0eba0d75c0, 0, 0x7f0eba0d7801, 3072) = 0
fileno(0x7f0eba0d75c0)                          = 2
__freading(0x7f0eba0d75c0, 0, 0x7f0eba0d7801, 3072) = 0
__freading(0x7f0eba0d75c0, 0, 0x7f0eba0d7801, 3072) = 0
fflush(0x7f0eba0d75c0)                          = 0
fclose(0x7f0eba0d75c0)                          = 0
+++ exited (status 0) +++
```

$ strace /bin/date
```
execve("/bin/date", ["/bin/date"], 0x7ffd8476f110 /* 39 vars */) = 0
brk(NULL)                               = 0x55dc21b8d000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
```

```
fstat(3, {st_mode=S_IFREG|0644, st_size=89954, ...}) = 0
mmap(NULL, 89954, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fa11b798000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0@n\2\0\0\0\0\0"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0755, st_size=1839792, ...}) = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa11b796000
mmap(NULL, 1852680, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fa11b5d1000
mprotect(0x7fa11b5f6000, 1662976, PROT_NONE) = 0
mmap(0x7fa11b5f6000, 1355776, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x25000) = 0x7fa11b5f6000
mmap(0x7fa11b741000, 303104, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x170000) =
0x7fa11b741000
mmap(0x7fa11b78c000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1ba000) = 0x7fa11b78c000
mmap(0x7fa11b792000, 13576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0)
= 0x7fa11b792000
close(3)                                = 0
arch_prctl(ARCH_SET_FS, 0x7fa11b797580) = 0
mprotect(0x7fa11b78c000, 12288, PROT_READ) = 0
mprotect(0x55dc2081a000, 4096, PROT_READ) = 0
mprotect(0x7fa11b7d8000, 4096, PROT_READ) = 0
munmap(0x7fa11b798000, 89954)           = 0
brk(NULL)                               = 0x55dc21b8d000
brk(0x55dc21bae000)                     = 0x55dc21bae000
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=3041456, ...}) = 0
mmap(NULL, 3041456, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fa11b2ea000
close(3)                                = 0
openat(AT_FDCWD, "/etc/localtime", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=3536, ...}) = 0
fstat(3, {st_mode=S_IFREG|0644, st_size=3536, ...}) = 0
read(3, "TZif2\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\5\0\0\0\5\0\0\0\0"..., 4096) = 3536
lseek(3, -2252, SEEK_CUR)               = 1284
read(3, "TZif2\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\5\0\0\0\5\0\0\0\0"..., 4096) = 2252
close(3)                                = 0
fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
write(1, "Sun 09 Jan 2022 04:35:32 AM EST\n", 32Sun 09 Jan 2022 04:35:32 AM EST
) = 32
close(1)                                = 0
close(2)                                = 0
exit_group(0)                           = ?
+++ exited with 0 +++
```

Executing /bin/date produces:

    System calls: 49
    Library calls: 45

   b)

Frequent system and library calls are:

1. pread64(): Reads up to count byte from file descriptor from given offset.
2. mmap(): Maps between files/devices and process address space
3. fstat(): Status of file

Problem 1.2: system call errors

a)
       i)   int open(const char *path, int oflag, ...)
- invalid assignment of arguments for oflag
- path provided doesn't exist or contains a file

      ii)   int close(int fildes)
- if fildes is not a valid file descriptor

b) Errno reports the last system call error that has occurred. So errno will be the error of the last failed system call.

Problem 1.3: execute a command in a modified environment or print the environment

Inside the folder you will find the code.