

作業系統 作業一

題目敘述

Write a C program that is passed an identifier for a shared-memory segment. This program will invoke the **shmctl()** function to obtain its **shm_ds structure**. It will then output the following values of the shared-memory segment:

- Segment ID
- Key
- Mode
- Owner UID
- Size
- Number of attaches

執行結果

本機測試

```
howard@howard-PE70-2QE: ~/SystemProgramming/Shared-Memory
howard@howard-PE70-2QE:~/SystemProgramming/Shared-Memory$ ll
total 36
drwxrwxr-x 2 howard howard 4096 11月  5 22:30 ./
drwxrwxr-x 8 howard howard 4096 11月  4 13:47 ../
-rwxrwxr-x 1 howard howard 8904 11月  5 22:29 a.out*
-rw-rw-r-- 1 howard howard  29 11月  5 22:30 makefile
-rw-rw-r-- 1 howard howard  127 11月  5 21:15 ref
-rw-rw-r-- 1 howard howard 1516 11月  5 22:19 shared_memory.c
-rw-rw-r-- 1 howard howard 1418 11月  5 21:31 SHM_fork.c
howard@howard-PE70-2QE:~/SystemProgramming/Shared-Memory$ gcc shared_memory.c
howard@howard-PE70-2QE:~/SystemProgramming/Shared-Memory$ ./a.out
Segment ID: 111116300
Key: 0
Mode: 384
Mode(rwx): 600
Owner UID: 1000
Size: 1024
Number of attaches: 0
howard@howard-PE70-2QE:~/SystemProgramming/Shared-Memory$
```

系上工作站測試(linux.cs.ccu.edu.tw)

```
howard@howard-PE70-2QE: ~/SystemProgramming/Shared-Memory
howard@howard-PE70-2QE:~/SystemProgramming/Shared-Memory$ scp shared_memory.c sch104u@linux.cs.ccu.edu.tw:~/
sch104u@linux.cs.ccu.edu.tw's password:
shared_memory.c 100% 1516 1.5KB/s 00:00
howard@howard-PE70-2QE:~/SystemProgramming/Shared-Memory$ ssh sch104u@linux.cs.ccu.edu.tw
sch104u@linux.cs.ccu.edu.tw's password:

System information as of Sun Nov  5 22:42:08 CST 2017

System load:  0.03          Processes:      253
Usage of /home: 13.1% of 21.58GB  Users logged in:  10
Memory usage:  3%          IP address for eth0: 140.123.101.4
Swap usage:    2%

Graph this data and manage this system at:
https://landscape.canonical.com/

151 packages can be updated.
69 updates are security updates.

===== CCU CSIE =====
Welcome to CCU CSIE Computing Server :-)          -last update:2017.09.22
SA group : popeye Setsal Alan Tako Joe
Computing Server:
  FreeBSD  csie0.cs.ccu.edu.tw
           csie1.cs.ccu.edu.tw
           csie2.cs.ccu.edu.tw ( www2.cs.ccu.edu.tw )
  Linux    linux.cs.ccu.edu.tw
           mcore8.cs.ccu.edu.tw
           000k0000G http://www.cs.ccu.edu.tw/lab401/
           0p00000000D0A00 mail 00 lab@cs.ccu.edu.tw 0p0000000G 23132
=====
Notice: this machine is based on amd64
Last login: Sun Nov  5 22:38:54 2017 from 36.236.53.172
sch104u@linux[10:42pm]~> gcc shared_memory.c
sch104u@linux[10:42pm]~> ./a.out
Segment ID: 103186432
Key: 0
Mode: 384
Mode(rwx): 600
Owner UID: 50215
Size: 1024
Number of attaches: 0
sch104u@linux[10:42pm]~> |
```

如何執行

1. 執行 **make** 或直直接執行編譯 **gcc hw1.c -o hw1**
2. 執行執行檔 **./hw1**
3. 程式將印出執行結果(如上圖)

其他說明

雖然作業需求上說不用把 mode 轉成 rwx 模式，直接把 int 印出來就好，不過我後來還是寫了一個 **decode function**，把 mode 傳給這個 function 之後，會把這個 mode 對應到 rwx 的 code 回傳回來。

```
// 把 mode 轉成 rwx 模式
char* decode(int x)
{
    int i, j, bit = 1, base, temp;
    char *ret = (char *) malloc(sizeof(char) * 4);
    ret[3] = '\0';

    for(i = 0; i < 3; i++)
    {
        temp = 0;
        base = 1;
        for(j = 0; j < 3; j++)
        {
            if(x & bit)
            {
                temp += base;
            }
            bit <<= 1;
            base <<= 1;
        }
        ret[2 - i] = temp + '0';
    }
    return ret;
}
```

呼叫完之後要記得在主程式碼把記憶體 free 掉。

```
char *rwxcode = decode((int) buf.shm_perm.mode);
printf("Mode(rwx): %s\n", rwxcode);
free(rwxcode);
```

執行結果

```
Mode: 384
Mode(rwx): 600
```