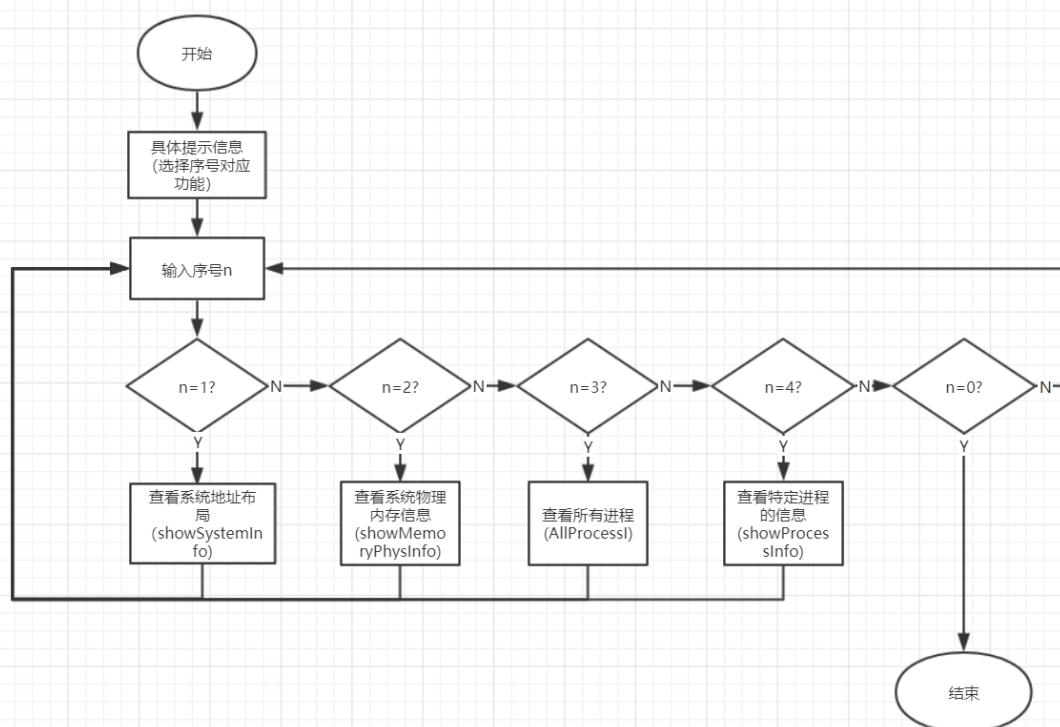


实验名称	内存监视		
学号	1120180207	姓名	唐小娟
<p>1. 实验目的</p> <p>1. 了解操作 Windows 的系统内存信息</p> <p>(1). 了解如何获取系统地址信息和物理内存信息</p> <p>(2). 了解如何获取当前内存使用情况</p> <p>(3). 了解如何获取当前进程信息</p> <p>(4). 了解如何获取特定进程的地址布局和工作集情况</p> <p>2. 了解操作 Linux 的系统内存信息。</p> <p>(1). 掌握 Linux 查看内存信息的有关命令</p> <p>二、实验内容</p> <p>1. Windows 系统</p> <p>(1). 实现内存监视器</p> <p>1). 设计流程</p> <p>2). 实行相关系统调用</p> <p>(2). 对实验二的 ParentProcess.cpp 进行相关修改</p> <p>2. Linux 系统</p> <p>(1). 理解并使用相关的命令</p> <p>三、实验环境及配置方法</p> <p>操作系统: Windows 10, Ubuntu 20.04, Linux 5.4.0-42</p> <p>集成开发环境: Microsoft VS Code, Microsoft Visual Studio 2019</p> <p>编译器: gcc 9.3.0</p> <p>四、实验方法和实验步骤（程序设计与实现）</p> <p>1. Windows 系统</p> <p>根据实验需要完成的功能，程序控制流程如下：</p>			



(1). ShowSystemInfo() 模块

调用 GetSystemInfo() 获取地址空间信息到 SYSTEM_INFO 类型 si 变量中，SYSTEM_INFO 结构体如下：

```

typedef struct _SYSTEM_INFO {
    union {
        DWORD dwOemId;
        struct {
            WORD wProcessorArchitecture;
            WORD wReserved;
        };
    };
    DWORD dwPageSize; //内存页的大小
    LPVOID lpMinimumApplicationAddress; //指向应用程序和动态链接库(DLL)
    //可以访问的最低内存地址。
    LPVOID lpMaximumApplicationAddress; //进程可用最大地址
    DWORD dwActiveProcessorMask;
    DWORD dwNumberOfProcessors; //处理机数量
    DWORD dwProcessorType;
    DWORD dwAllocationGranularity; //虚拟内存的粒度
    WORD wProcessorLevel;
    WORD wProcessorRevision;
} SYSTEM_INFO;
    
```

(2). showMemoryPhysInfo() 模块

操作系统课程设计实验报告

调用 `GlobalMemoryStatusEx()` 获取内存信息到 `MEMORYSTATUSEX` 结构体类型 `memPhys` 变量中, `MEMORYSTATUSEX` 结构体如下:

```
typedef struct _MEMORYSTATUSEX {
    DWORD dwLength;
    DWORD dwMemoryLoad;           //内存占用率
    DWORDLONG ullTotalPhys;       //总物理内存
    DWORDLONG ullAvailPhys;       //剩余物理内存
    DWORDLONG ullTotalPageFile;
    DWORDLONG ullAvailPageFile;
    DWORDLONG ullTotalVirtual;    //虚拟内存总量
    DWORDLONG ullAvailVirtual;    //虚拟内存剩余量
    DWORDLONG ullAvailExtendedVirtual;
} MEMORYSTATUSEX, *LPMEMORYSTATUSEX;
```

调用 `GetPerformanceInfo()` 获取信息到 `PERFORMANCE_INFORMATION` 结构体 `pi` 变量中, `PERFORMANCE_INFORMATION` 结构体如下:

```
typedef struct _PERFORMANCE_INFORMATION {
    DWORD cb;
    SIZE_T CommitTotal;           //系统当前提交的页面总数
    SIZE_T CommitLimit;
    SIZE_T CommitPeak;
    SIZE_T PhysicalTotal;         //全部物理内存
    SIZE_T PhysicalAvailable;     //可用物理内存
    SIZE_T SystemCache;           //系统缓存
    SIZE_T KernelTotal;           //内核占内存总大小(当前分页和非分页内核池中的内存总和)
    SIZE_T KernelPaged;           //当前分页内核池中的内存
    SIZE_T KernelNonpaged;        //当前在非分页内核池中的内存
    SIZE_T PageSize;
    DWORD HandleCount;            //句柄计数
    DWORD ProcessCount;           //进程计数
    DWORD ThreadCount;            //线程计数
} PERFORMANCE_INFORMATION;
```

(3). `AllProcessID()` 模块

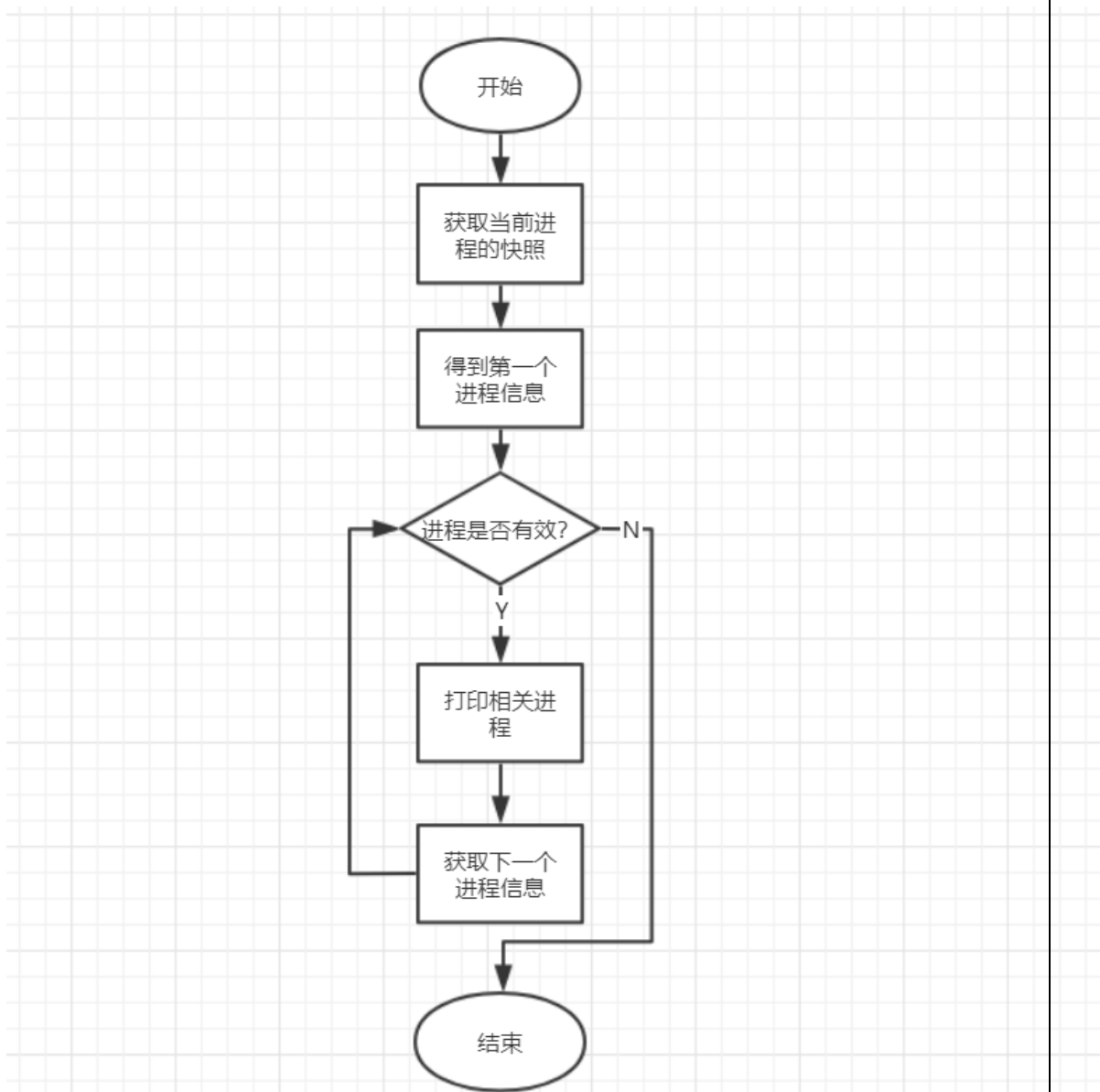
调用 `CreateToolhelp32Snapshot()` 获取当前进程的快照, 再调用 `Process32First()` 获取第一个进程的信息, 之后循环调用 `Process32Next` 得到所有进程的信息, 包括进程 `pid`, 进程名字, 优先级。相关函数信息:

```
HANDLE WINAPI CreateToolhelp32Snapshot(
```

操作系统课程设计实验报告

```
DWORD dwFlags,          //用来指定“快照”中需要返回的对象
DWORD th32ProcessID      //进程 ID 号
);
```

流程图如下：



(4). showProcessInfo() 模块

调用 `OpenProcess()` 打开一个已存在的进程对象，并返回进程的句柄 `hProcess`；再调用 `GetProcessMemoryInfo()` 得到工作集大小，之后进入循环调用 `VirtualQueryEx` 查找虚拟内存块信息 `MEMORY_BASIC_INFORMATION mbi`，包括块的长度、虚拟内存块的状态、保护、类型以及模块名。相关函数和结构体信息如下：

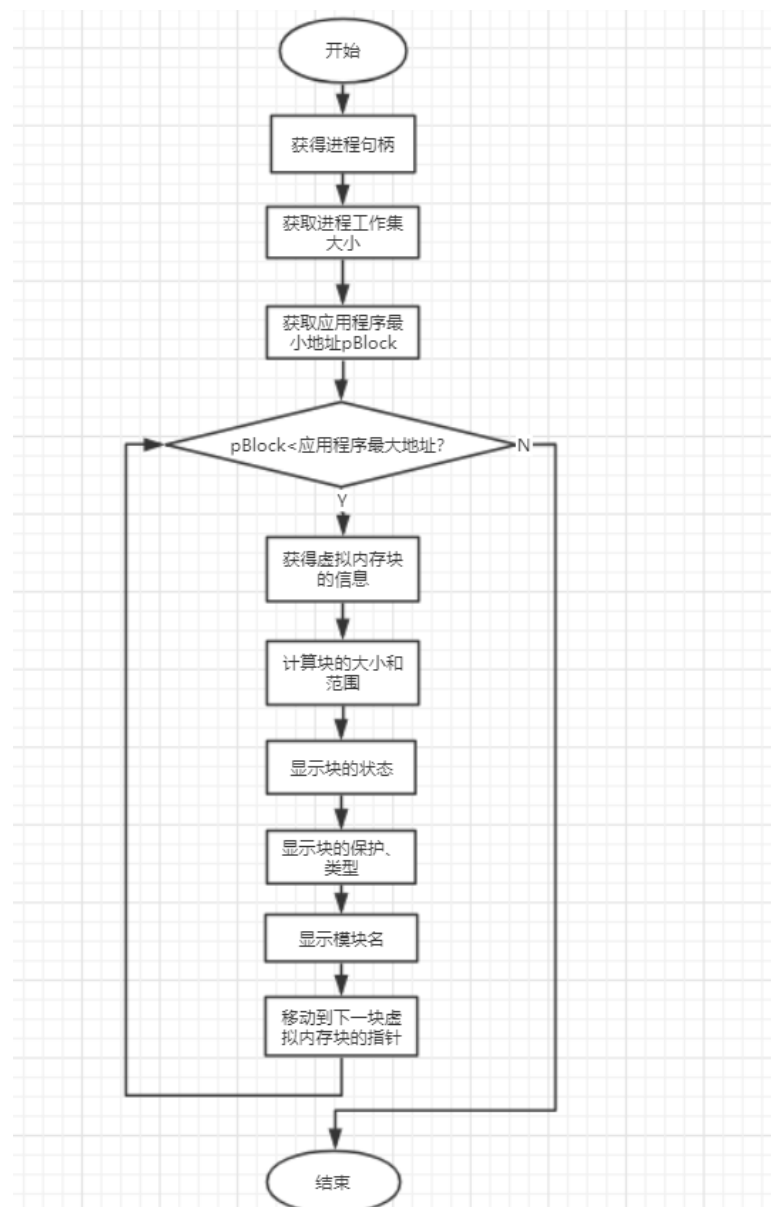
```
HANDLE OpenProcess(
    DWORD dwDesiredAccess, //渴望得到的访问权限（标志）
    BOOL bInheritHandle,   // 是否继承句柄
```

操作系统课程设计实验报告

```
        DWORD dwProcessId      // 进程标示符
    );

typedef struct _MEMORY_BASIC_INFORMATION {
    PVOID BaseAddress;          // 区域基地址。
    PVOID AllocationBase;       // 分配基地址。
    DWORD AllocationProtect;    // 区域被初次保留时赋予的保护属性。
    SIZE_T RegionSize;          // 区域大小（以字节为计量单位）。
    DWORD State;                // 状态（MEM_FREE、MEM_RESERVE 或 MEM_COMMIT）。
    DWORD Protect;              // 保护属性。
    DWORD Type;                 // 类型。
} MEMORY_BASIC_INFORMATION;
```

流程图如下：



2. Linux 系统

(1). top 命令

用来监控 linux 的系统状况，默认 5s 页面更新一次。运行结果如下：

```
ostxj@ostxj-virtual-machine: ~/Desktop
top - 16:13:52 up 1 min, 1 user, load average: 0.47, 0.20, 0.08
Tasks: 386 total, 1 running, 385 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.0 us, 0.7 sy, 0.0 ni, 98.2 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 3907.3 total, 1589.8 free, 1109.4 used, 1208.1 buff/cache
MiB Swap: 1873.4 total, 1873.4 free, 0.0 used, 2553.7 avail Mem

  PID USER      PR  NI   VIRT    RES    SHR  S  %CPU  %MEM    TIME+  COMMAND
 2091 ostxj     20   0 4073848 248596 100956 S   6.3   6.2   0:04.62 gnome-shell
 1865 ostxj     20   0 268152  60172  34128 S   5.3   1.5   0:01.67 Xorg
 2324 ostxj     20   0 1249912 203500  44128 S   1.0   5.1   0:04.05 snap-store
 342  root      0 -20    0         0      0 I   0.3   0.0   0:00.05 kworker/1:1H-kblockd
 395  root      0 -20    0         0      0 I   0.3   0.0   0:00.02 kworker/0:1H-kblockd
 777  systemd+  20   0  24320  12384   8136 S   0.3   0.3   0:00.09 systemd-resolve
 831  root      20   0 1669196 38856  15528 S   0.3   1.0   0:01.66 snapd
2180 ostxj     20   0 431912  31384  21052 S   0.3   0.8   0:00.32 gsd-color
2187 ostxj     20   0 912520  33408  22636 S   0.3   0.8   0:00.33 gsd-media-keys
2210 ostxj     20   0 358348  32188  21416 S   0.3   0.8   0:00.35 gsd-xsettings
2242 ostxj     20   0 216504  28336  25052 S   0.3   0.7   0:00.28 vmtotlsd
2516 root      20   0 386364  24448  20820 S   0.3   0.6   0:00.12 fwupd
2606 ostxj     20   0 1012440 56652  42580 S   0.3   1.4   0:00.54 gnome-terminal-
2627 ostxj     20   0  20764   4308   3512 R   0.3   0.1   0:00.08 top
   1  root      20   0  167780  11648   8428 S   0.0   0.3   0:09.71 systemd
   2  root      20   0    0         0      0 S   0.0   0.0   0:00.02 kthreadd
   3  root      0 -20    0         0      0 I   0.0   0.0   0:00.00 rcu_gp
   4  root      0 -20    0         0      0 I   0.0   0.0   0:00.00 rcu_par_gp
   5  root      20   0    0         0      0 I   0.0   0.0   0:00.00 kworker/0:0-events
   6  root      0 -20    0         0      0 I   0.0   0.0   0:00.00 kworker/0:0H-kblockd
   7  root      20   0    0         0      0 I   0.0   0.0   0:00.00 kworker/0:1-rcu_par_gp
   8  root      20   0    0         0      0 I   0.0   0.0   0:00.00 kworker/u256:0-events_freezable_power_
   9  root      0 -20    0         0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
  10  root      20   0    0         0      0 S   0.0   0.0   0:00.02 ksoftirqd/0
  11  root      20   0    0         0      0 I   0.0   0.0   0:00.08 rcu_sched
  12  root      rt   0    0         0      0 S   0.0   0.0   0:00.00 migration/0
  13  root     -51   0    0         0      0 S   0.0   0.0   0:00.00 idle_inject/0
  14  root      20   0    0         0      0 S   0.0   0.0   0:00.00 cpuhp/0
  15  root      20   0    0         0      0 S   0.0   0.0   0:00.00 cpuhp/1
```

第一行：

16: 13: 52	当前时间
up 1 min	系统运行时间
1 user	当前登录用户数
load average: 0.47, 0.20, 0.08	系统 1 分钟、5 分钟、15 分钟负载情况

第二行：进程情况

386 total	进程总数
1 running	正在运行的进程数
385 sleeping	睡眠的进程数
0 stopped	停止的进程数
0 zombie	僵尸进程数

第三行：CPU 的信息

1.0 us	用户空间占用 CPU 百分比
--------	----------------

操作系统课程设计实验报告

0.7 sy	内核空间占用 CPU 百分比
0.0 ni	用户进程空间内改变过优先级的进程占用 CPU 百分比
98.2 id	空闲 CPU 百分比
0.0 wa	等待输入输出的 CPU 时间百分比
0.0 hi	硬中断占用 CPU 的百分比
0.1 si	软中断占用 CPU 的百分比
0.0 st	用于有虚拟 cpu 的情况, 用来指示被虚拟机偷掉的 cpu 时间
第四行: 物理内存信息	
3907.3 total	物理内存总量
1589.8 free	空闲内存总量
1109.4 used	使用的物理内存总量
1208.1 buff/cache	用作内核缓存的内存量
第五行: 交换区信息	
1873.4 total	交换区总量
1873.4 free	空闲交换区总量
0.0 used	使用的交换区总量
2553.7 avail mem	代表可用于进程下一次分配的物理内存数量
第六行: 相关进程信息	
PID	进程 id
USER	进程所有者的用户名
PR	优先级
NI	nice 值。负值表示高优先级, 正值表示低优先级
VIRT	进程使用的虚拟内存总量, 单位 kb。 VIRT=SWAP+RES
RES	进程使用的、未被换出的物理内存大小,

操作系统课程设计实验报告

	单位 kb。RES=CODE+DATA
SHR	共享内存大小，单位 kb
S	进程状态。D=不可中断的睡眠状态 R=运行 S=睡眠 T=跟踪/停止 Z=僵尸进程
%CPU	上次更新到现在的 CPU 时间占用百分比
%MEM	进程使用的物理内存百分比
TIME+	进程使用的 CPU 时间总计，单位 1/100 秒
COMMAND	命令名/命令行

子命令参数含义：

T：根据时间/累计时间 (TIME+) 进行从大到小排序

M：根据驻留内存 (%MEM) 大小进行排序

P：根据 CPU 使用百分比 (%CPU) 大小进行排序

(2). ps -A 命令

查看所有进程。

(3). top -p pid 命令

查看相应 pid 的进程信息。

(4). pmap -d pid 命令

查看相应 pid 的进程内存使用情况。

Address	映像起始地址
Kbytes	映像大小 (KB)
Mode	映像权限
Offset	文件偏移
Device	设备名
Mapping	映像支持文件

最后一行：

mapped	该进程映射的虚拟地址空间大小
writable/private	表示进程所占用的私有地址空间大小
shared	表示进程和其他进程共享的内存大小

五、实验结果和分析

1. Windows 系统

(1). 系统地址空间布局

```
*****内存监视器*****
序号对应的功能：
1. 系统内存地址空间布局
2. 系统物理内存信息
3. 查看所有进程
4. 查看特定进程的信息
0. 退出
请输入序号：1
*****系统内存地址空间布局*****
内存页的大小：4KB
进程可用最小地址：0000000000010000
进程可用最大地址：00007FFFFFFEFFFF
处理机数量：16
虚拟内存的粒度：64KB

显示完毕！
```

(2). 显示物理内存信息

```
请输入序号：2
*****系统物理内存信息*****
物理内存总量：15.8699GB
物理内存使用率：77%
物理内存使用量：12.3109GB
虚拟内存总量：131072GB
虚拟内存使用量：4143.34MB
系统当前提交的页面总数：568857页
系统缓存内存量：940789页
当前分页和非分页内核池中的内存总和：315750页
当前分页内核池中的内存：160719页
当前在非分页内核池中的内存：155031页
打开句柄数：128459
当前进程数：296
当前线程数：3509

显示完毕！
```

(3). 显示所有进程

操作系统课程设计实验报告

请输入序号: 3

***** 查看所有进程的相关信息*****

序号	PID	名称	优先级
1	0	[System Process]	0
2	4	System	8
3	172	Registry	8
4	504	smss.exe	11
5	784	csrss.exe	13
6	856	wininit.exe	13
7	864	csrss.exe	13
8	948	services.exe	9
9	964	lsass.exe	9
10	548	svchost.exe	8
11	1032	svchost.exe	8
12	1048	fontdrvhost.exe	8
13	1088	WUDFHost.exe	13
14	1144	svchost.exe	8
15	1196	svchost.exe	8
16	1248	WUDFHost.exe	8
17	1308	winlogon.exe	13
18	1380	fontdrvhost.exe	8
19	1452	dwm.exe	13
20	1512	svchost.exe	8
21	1528	svchost.exe	8
22	1556	svchost.exe	8
23	1564	svchost.exe	8
24	1572	svchost.exe	8
25	1724	svchost.exe	8
269	13836	MSBuild.exe	8
270	6284	conhost.exe	8
271	19484	mspdbsrv.exe	8
272	6840	msvsmon.exe	8
273	17856	StandardCollector.Service.exe	8
274	12816	VsDebugConsole.exe	8
275	20312	conhost.exe	8
276	13764	memorymonitor.exe	8
277	21380	msvsmon.exe	8
278	15680	ScriptedSandbox64.exe	8
279	19020	Code.exe	4
280	6700	conhost.exe	8
281	20984	powershell.exe	8
282	20228	Code.exe	8
283	20416	CodeHelper.exe	8
284	11224	conhost.exe	8
285	13156	cpptools.exe	8
286	20224	conhost.exe	8
287	16124	cpptools-srv.exe	8
288	13772	Code.exe	8
289	7820	WmiPrvSE.exe	8
290	7248	cpptools-srv.exe	8
291	4672	SearchProtocolHost.exe	4
292	14924	SearchFilterHost.exe	4
293	13940	svchost.exe	8
294	13724	ParentProcess.exe	8

显示完毕!

(4). 显示 ParentProcess 进程信息

根据所有进程可得 ParentProcess.exe 的进程 pid, 可以得到该进程的相关信息

```
294 13724 ParentProcess.exe 8
显示完毕!
请输入序号: 4
请输入进程号pid: 13724
*****查看特定进程的信息*****
当前进程的工作集大小: 4.35938MB
块的范围 状态 保护类型 文件类型 模块名
00010000-00020000 ( 64.0 KB) 提交 READWRITE Mapped
00020000-00021000 ( 4.00 KB) 提交 READONLY Mapped
00021000-00030000 ( 60.0 KB) 空闲 NOACCESS
00030000-0004d000 ( 116 KB) 提交 READONLY Mapped
0004d000-00050000 ( 12.0 KB) 空闲 NOACCESS
00050000-00054000 ( 16.0 KB) 提交 READONLY Mapped
00054000-00060000 ( 48.0 KB) 空闲 NOACCESS
00060000-00062000 ( 8.00 KB) 提交 READWRITE Private
00062000-00070000 ( 56.0 KB) 空闲 NOACCESS
00070000-00139000 ( 804 KB) 提交 READONLY Mapped
00139000-00140000 ( 28.0 KB) 空闲 NOACCESS
00140000-00141000 ( 4.00 KB) 提交 READONLY Mapped
00141000-00150000 ( 60.0 KB) 空闲 NOACCESS
00150000-00151000 ( 4.00 KB) 提交 READONLY Mapped
00151000-00160000 ( 60.0 KB) 空闲 NOACCESS
00160000-00162000 ( 8.00 KB) 提交 READWRITE Private
00162000-001c2000 ( 384 KB) 保留 READONLY Private
001c2000-001d0000 ( 56.0 KB) 空闲 NOACCESS
001d0000-001d4000 ( 16.0 KB) 提交 READONLY Mapped
001d4000-001d8000 ( 16.0 KB) 保留 READONLY Mapped
001d8000-00200000 ( 160 KB) 空闲 NOACCESS
00200000-002b4000 ( 720 KB) 保留 READONLY Private
002b4000-002b7000 ( 12.0 KB) 提交 READWRITE Private
002b7000-00400000 ( 1.28 MB) 保留 READONLY Private
00400000-00401000 ( 4.00 KB) 提交 READONLY Image
00401000-00404000 ( 12.0 KB) 提交 EXECUTE_READ Image
00404000-00405000 ( 4.00 KB) 提交 READWRITE Image
00405000-00406000 ( 4.00 KB) 提交 READONLY Image
00406000-00408000 ( 8.00 KB) 提交 EXECUTE_WRITECOPY Image
00408000-0040a000 ( 8.00 KB) 提交 READWRITE Image
0040a000-0040c000 ( 8.00 KB) 提交 WRITECOPY Image
0040c000-00411000 ( 20.0 KB) 提交 READONLY Image
00411000-00420000 ( 60.0 KB) 空闲 NOACCESS
00420000-00619000 ( 1.97 MB) 保留 READONLY Private
00619000-0061c000 ( 12.0 KB) 提交 GUARD Private
0061c000-00620000 ( 16.0 KB) 提交 READWRITE Private
00620000-00621000 ( 4.00 KB) 提交 READWRITE Private
00621000-00682000 ( 388 KB) 保留 READONLY Private
```

3. Linux 操作系统

(1). top 命令

操作系统课程设计实验报告

```
top - 16:55:47 up 43 min, 1 user, load average: 0.06, 0.13, 0.07
Tasks: 350 total, 1 running, 349 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.8 sy, 0.0 ni, 99.2 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3907.3 total, 1284.1 free, 1342.9 used, 1280.2 buff/cache
MiB Swap: 1873.4 total, 1873.4 free, 0.0 used, 2315.8 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	167780	11648	8428	S	0.0	0.3	0:09.78	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.02	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kblockd
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
10	root	20	0	0	0	0	S	0.0	0.0	0:00.02	ksoftirqd/0
11	root	20	0	0	0	0	I	0.0	0.0	0:01.15	rcu_sched
12	root	rt	0	0	0	0	S	0.0	0.0	0:00.01	migration/0
13	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/0
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
16	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/1
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/1
18	root	20	0	0	0	0	S	0.0	0.0	0:00.01	ksoftirqd/1
20	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H-kblockd
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/2
22	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/2
23	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/2
24	root	20	0	0	0	0	S	0.0	0.0	0:00.01	ksoftirqd/2
26	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/2:0H-kblockd
27	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/3
28	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/3
29	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/3
30	root	20	0	0	0	0	S	0.0	0.0	0:00.02	ksoftirqd/3
32	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/3:0H-kblockd
33	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/4
34	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/4
35	root	rt	0	0	0	0	S	0.0	0.0	0:00.49	migration/4

输入参数 P:

```
top - 16:56:32 up 44 min, 1 user, load average: 0.03, 0.11, 0.07
Tasks: 350 total, 1 running, 349 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.1 us, 0.2 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3907.3 total, 1283.9 free, 1343.1 used, 1280.3 buff/cache
MiB Swap: 1873.4 total, 1873.4 free, 0.0 used, 2315.7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1865	ostxj	20	0	291736	66544	34752	S	2.0	1.7	0:19.99	Xorg
2418	ostxj	20	0	272760	22948	17356	S	0.7	0.6	0:00.90	ibus-engine-lib
2606	ostxj	20	0	1013860	58944	43708	S	0.7	1.5	0:07.38	gnome-terminal-
5248	ostxj	20	0	20752	4412	3456	R	0.7	0.1	0:00.09	top
1	root	20	0	167780	11648	8428	S	0.0	0.3	0:09.78	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.02	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kblockd
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
10	root	20	0	0	0	0	S	0.0	0.0	0:00.02	ksoftirqd/0
11	root	20	0	0	0	0	I	0.0	0.0	0:01.17	rcu_sched
12	root	rt	0	0	0	0	S	0.0	0.0	0:00.01	migration/0
13	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/0
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
16	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/1
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/1
18	root	20	0	0	0	0	S	0.0	0.0	0:00.01	ksoftirqd/1
20	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H-kblockd
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/2
22	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/2
23	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/2
24	root	20	0	0	0	0	S	0.0	0.0	0:00.01	ksoftirqd/2
26	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/2:0H-kblockd
27	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/3
28	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/3
29	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/3
30	root	20	0	0	0	0	S	0.0	0.0	0:00.02	ksoftirqd/3

由上可知，进程是根据 CPU 使用百分比(%CPU)大小进行从大到小排序的。

操作系统课程设计实验报告

输入参数 T:

```
top - 16:56:03 up 43 min, 1 user, load average: 0.05, 0.12, 0.07
Tasks: 350 total, 1 running, 349 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.2 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3907.3 total, 1284.1 free, 1342.9 used, 1280.3 buff/cache
MiB Swap: 1873.4 total, 1873.4 free, 0.0 used, 2315.9 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2091	ostxj	20	0	4335996	280348	101556	S	1.4	7.0	0:21.92	gnome-shell
1865	ostxj	20	0	291736	66544	34752	S	2.1	1.7	0:19.90	Xorg
1	root	20	0	167780	11648	8428	S	0.0	0.3	0:09.78	systemd
2606	ostxj	20	0	1013860	58944	43708	S	1.0	1.5	0:07.33	gnome-terminal-
2324	ostxj	20	0	1300608	325396	43224	S	0.0	8.1	0:06.58	snap-store
4897	ostxj	20	0	1202776	71548	48300	S	0.0	1.8	0:03.83	nautilus
1972	ostxj	20	0	326332	13980	7864	S	0.7	0.3	0:02.80	ibus-daemon
810	root	20	0	347640	22416	18800	S	0.0	0.6	0:02.77	NetworkManager
2242	ostxj	20	0	225892	28336	25052	S	0.0	0.7	0:02.53	vmtoolsd
1242	root	20	0	169596	13756	11276	S	0.0	0.3	0:02.03	vmtoolsd
831	root	20	0	1669196	29368	15568	S	0.0	0.7	0:01.78	snapd
1322	root	20	0	489592	10152	8600	S	0.0	0.3	0:01.28	ManagementAgent
1980	ostxj	20	0	405572	62264	39524	S	0.0	1.6	0:01.24	ibus-extension-
11	root	20	0	0	0	0	I	0.0	0.0	0:01.16	rcu sched
826	root	20	0	253916	14044	9700	S	0.0	0.4	0:01.06	polkitd
1773	ostxj	9	-11	1753100	19688	15436	S	0.0	0.5	0:01.05	pulseaudio
2418	ostxj	20	0	272760	22948	17356	S	0.0	0.6	0:00.89	ibus-engine-lib
808	message+	20	0	9876	6480	3960	S	0.0	0.2	0:00.76	dbus-daemon
1779	ostxj	20	0	13564	10696	3972	S	0.0	0.3	0:00.54	dbus-daemon
53	root	rt	0	0	0	0	S	0.0	0.0	0:00.50	migration/7
35	root	rt	0	0	0	0	S	0.0	0.0	0:00.49	migration/4
41	root	rt	0	0	0	0	S	0.0	0.0	0:00.49	migration/5
803	avahi	20	0	8508	3520	3188	S	0.0	0.1	0:00.49	avahi-daemon
2031	ostxj	20	0	507440	33384	22276	S	0.0	0.8	0:00.49	xdg-desktop-por
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/1
23	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/2
29	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/3
47	root	rt	0	0	0	0	S	0.0	0.0	0:00.48	migration/6
2013	ostxj	20	0	162820	7720	6944	S	0.3	0.2	0:00.47	at-spi2-registr

由上可知，进程是根据时间/累计时间 (TIME+) 进行从大到小排序的。

输入参数 M:

```
ostxj@ostxj-virtual-machine: ~/Desktop
top - 16:56:49 up 44 min, 1 user, load average: 0.02, 0.10, 0.07
Tasks: 350 total, 1 running, 349 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.2 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3907.3 total, 1282.2 free, 1344.8 used, 1280.3 buff/cache
MiB Swap: 1873.4 total, 1873.4 free, 0.0 used, 2314.0 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2324	ostxj	20	0	1300608	325396	43224	S	0.0	8.1	0:06.58	snap-store
2091	ostxj	20	0	4335996	280352	101556	S	0.7	7.0	0:22.05	gnome-shell
1979	ostxj	20	0	447836	72736	52724	S	0.0	1.8	0:00.31	ibus-ui-gtk3
4897	ostxj	20	0	1202776	71548	48300	S	0.0	1.8	0:03.83	nautilus
1865	ostxj	20	0	291736	66544	34752	S	2.3	1.7	0:20.14	Xorg
1980	ostxj	20	0	405572	62264	39524	S	0.0	1.6	0:01.24	ibus-extension-
2238	ostxj	20	0	647844	61416	45608	S	0.0	1.5	0:00.32	evolution-alarm
2606	ostxj	20	0	1013860	58944	43708	S	1.7	1.5	0:07.47	gnome-terminal-
1983	ostxj	20	0	327644	57212	39792	S	0.0	1.4	0:00.36	ibus-x11
2516	root	20	0	408272	44676	19152	S	0.0	1.1	0:00.34	fwupd
1833	ostxj	20	0	555072	36848	30816	S	0.0	0.9	0:00.03	goa-daemon
3295	ostxj	20	0	432080	34072	24388	S	0.0	0.9	0:00.22	update-notifier
2187	ostxj	20	0	912520	33408	22636	S	0.0	0.8	0:00.37	gsd-media-keys
2031	ostxj	20	0	507440	33384	22276	S	0.0	0.8	0:00.49	xdg-desktop-por
2210	ostxj	20	0	358348	32248	21464	S	0.0	0.8	0:00.39	gsd-xsettings
2190	ostxj	20	0	357732	31672	21284	S	0.0	0.8	0:00.38	gsd-power
2180	ostxj	20	0	431912	31512	21180	S	0.0	0.8	0:00.40	gsd-color
2186	ostxj	20	0	357152	30884	20764	S	0.0	0.8	0:00.35	gsd-keyboard
2138	ostxj	20	0	847616	30196	26380	S	0.0	0.8	0:00.07	evolution-calen
2206	ostxj	20	0	356484	29984	19784	S	0.0	0.7	0:00.35	gsd-wacom
2149	ostxj	20	0	682256	29776	26064	S	0.0	0.7	0:00.03	evolution-addre
831	root	20	0	1669196	29368	15568	S	0.0	0.7	0:01.79	snapd
2242	ostxj	20	0	225892	28336	25052	S	0.3	0.7	0:02.59	vmtoolsd
422	root	19	-1	84960	27952	26276	S	0.0	0.7	0:00.46	systemd-journal
2160	ostxj	20	0	2939136	27448	22096	S	0.0	0.7	0:00.05	gjs
1775	ostxj	39	19	594908	26636	16680	S	0.0	0.7	0:00.45	tracker-miner-f
2131	ostxj	20	0	399248	26068	22248	S	0.0	0.7	0:00.04	evolution-sourc
2418	ostxj	20	0	272760	22948	17356	S	0.0	0.6	0:00.90	ibus-engine-lib
945	root	20	0	126464	22816	14712	S	0.0	0.6	0:00.09	unattended-upgr

由上可知，进程是根据驻留内存 (%MEM) 大小进行从大到小排序的。

(2). ps -A 命令

```
PID TTY          TIME CMD
  1 ?           00:00:09 systemd
  2 ?           00:00:00 kthreadd
  3 ?           00:00:00 rcu_gp
  4 ?           00:00:00 rcu_par_gp
  6 ?           00:00:00 kworker/0:0H-kblockd
  9 ?           00:00:00 mm_percpu_wq
 10 ?           00:00:00 ksoftirqd/0
 11 ?           00:00:01 rcu_sched
 12 ?           00:00:00 migration/0
 13 ?           00:00:00 idle_inject/0
 14 ?           00:00:00 cpuhp/0
 15 ?           00:00:00 cpuhp/1
 16 ?           00:00:00 idle_inject/1
 17 ?           00:00:00 migration/1
 18 ?           00:00:00 ksoftirqd/1
 20 ?           00:00:00 kworker/1:0H-kblockd
 21 ?           00:00:00 cpuhp/2
 22 ?           00:00:00 idle_inject/2
 23 ?           00:00:00 migration/2
 24 ?           00:00:00 ksoftirqd/2
 26 ?           00:00:00 kworker/2:0H-kblockd
 27 ?           00:00:00 cpuhp/3
 28 ?           00:00:00 idle_inject/3
 29 ?           00:00:00 migration/3
 30 ?           00:00:00 ksoftirqd/3
 32 ?           00:00:00 kworker/3:0H-kblockd
 33 ?           00:00:00 cpuhp/4
 34 ?           00:00:00 idle_inject/4
 35 ?           00:00:00 migration/4
 36 ?           00:00:00 ksoftirqd/4
 38 ?           00:00:00 kworker/4:0H-kblockd
 39 ?           00:00:00 cpuhp/5
 40 ?           00:00:00 idle_inject/5
 41 ?           00:00:00 migration/5
```

```

2238 ?          00:00:00 evolution-alarm
2242 ?          00:00:02 vmttoolsd
2307 ?          00:00:00 gsd-printer
2324 ?          00:00:06 snap-store
2418 ?          00:00:00 ibus-engine-lib
2501 ?          00:00:00 gvfsd-trash
2516 ?          00:00:00 fwupd
2606 ?          00:00:08 gnome-terminal-
3066 ?          00:00:00 zeitgeist-datah
3078 ?          00:00:00 zeitgeist-daemo
3085 ?          00:00:00 zeitgeist-fts
3292 ?          00:00:00 gvfsd-metadata
3295 ?          00:00:00 update-notifier
4869 ?          00:00:00 kworker/u256:0-ext4-rsv-conversion
4894 ?          00:00:00 kworker/1:0-cgroup_destroy
4897 ?          00:00:04 nautilus
4915 ?          00:00:00 kworker/3:2-events
4916 ?          00:00:00 kworker/0:0-events
4936 ?          00:00:00 kworker/2:0-mm_percpu_wq
4940 ?          00:00:00 kworker/7:0-events
4941 ?          00:00:00 kworker/4:0-events
4956 pts/1      00:00:00 bash
4962 ?          00:00:00 kworker/5:0-mm_percpu_wq
5156 ?          00:00:00 kworker/u256:2-events_unbound
5199 ?          00:00:00 kworker/5:1-events
5206 ?          00:00:00 kworker/4:2-events
5226 ?          00:00:00 kworker/7:2-events
5241 ?          00:00:00 kworker/6:2-mm_percpu_wq
5292 ?          00:00:00 kworker/1:1-events
5299 ?          00:00:00 kworker/u256:1-events_unbound
5312 ?          00:00:00 kworker/5:2-events
5327 ?          00:00:00 kworker/3:0-events
5329 pts/1      00:00:00 ParentProcess.e
5351 pts/0       00:00:00 bash
5357 pts/0       00:00:00 ps
  
```

根据上图可得到 ParentProcess.exe 的进程 pid

(3). top -p 5329

```

top - 17:02:07 up 49 min, 1 user, load average: 0.00, 0.03, 0.04
Tasks: 1 total, 0 running, 1 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.8 sy, 0.0 ni, 99.2 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3907.3 total, 1273.2 free, 1351.5 used, 1282.6 buff/cache
MiB Swap: 1873.4 total, 1873.4 free, 0.0 used, 2307.1 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
  5329 ostxj    20   0   2496   708   644 S   0.0   0.0   0:00.00 ParentProcess.e
  
```

由上图，可以看到有一个进程正处于 sleeping

(4). pmap -d 5329

```

ostxj@ostxj-virtual-machine:~$ pmap -d 5405
5405: ./ParentProcess.exe ChildProcess
Address      Kbytes Mode  Offset      Device      Mapping
000055f55b5b2000    4 r---- 0000000000000000 008:00005 ParentProcess.exe
000055f55b5b3000    4 r-x-- 00000000000001000 008:00005 ParentProcess.exe
000055f55b5b4000    4 r---- 00000000000002000 008:00005 ParentProcess.exe
000055f55b5b5000    4 r---- 00000000000002000 008:00005 ParentProcess.exe
000055f55b5b6000    4 rw--- 00000000000003000 008:00005 ParentProcess.exe
000055f55b7fa000   132 rw--- 0000000000000000 000:00000 [ anon ]
00007feb9f767000   148 r---- 0000000000000000 008:00005 libc-2.31.so
00007feb9f78c000  1504 r-x-- 00000000000025000 008:00005 libc-2.31.so
00007feb9f904000   296 r---- 0000000000019d000 008:00005 libc-2.31.so
00007feb9f94e000    4 ----- 000000000001e7000 008:00005 libc-2.31.so
00007feb9f94f000   12 r---- 000000000001e7000 008:00005 libc-2.31.so
00007feb9f952000   12 rw--- 000000000001ea000 008:00005 libc-2.31.so
00007feb9f955000   24 rw--- 0000000000000000 000:00000 [ anon ]
00007feb9f96e000    4 r---- 0000000000000000 008:00005 ld-2.31.so
00007feb9f96f000   140 r-x-- 00000000000001000 008:00005 ld-2.31.so
00007feb9f992000   32 r---- 00000000000024000 008:00005 ld-2.31.so
00007feb9f99b000    4 r---- 0000000000002c000 008:00005 ld-2.31.so
00007feb9f99c000    4 rw--- 0000000000002d000 008:00005 ld-2.31.so
00007feb9f99d000    4 rw--- 0000000000000000 000:00000 [ anon ]
00007ffe66a57000   132 rw--- 0000000000000000 000:00000 [ stack ]
00007ffe66ba2000   16 r---- 0000000000000000 000:00000 [ anon ]
00007ffe66ba6000    8 r-x-- 0000000000000000 000:00000 [ anon ]
fffffffff600000    4 --x-- 0000000000000000 000:00000 [ anon ]
mapped: 2500K    writeable/private: 312K    shared: 0K

```

六、讨论、心得