

# 同济大学计算机系

## 操作系统实验报告



学 号 2152809

姓 名 曾崇然

专 业 计算机科学与技术

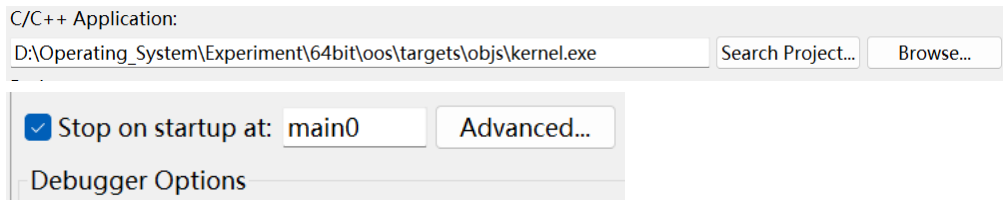
授课老师 方钰老师

## 一. 完成实验准备

- (1) 我在本次实验中采用的是之前实验二中的可运行程序 showStack.exe

```
[/]#showStack.exe
result=3
```

- (2) 设置调试对象和调试的起点



## 二. 找到完整的进程图像 (题目(1))

- (1) 调试运行

设置断点并运行至断点处

```
Diagnose::Write("Process %d is exiting\n",u.u_procp->p_pid);
/* Reset Tracing flag */
u.u_procp->p_flag &= (~Process::STRC);
```

- (2) 获取进程的 user 结构

Address	0 - 3	4 - 7	8 - B	C - F
C03FF000	8CFF3FC0	A4FF3FC0	00000000	00000000
C03FF010	009611C0	008020C0	00104000	00300000
C03FF020	00404000	00300000	00100000	DCFF3FC0

变量名称	含义	值
Process* u_procp	Proc 结构的逻辑地址	0xC0119600
MemoryDescriptor u_MemoryDescriptor (定义如下, 此处均为逻辑地址)		
PageTable* m_UserPageTableArray	相对映射表首地址	0xC0208000
unsigned long m_TextStartAddress	代码段起始地址	0x00401000=4M+4K
unsigned long m_TextSize	代码段长度	0x00003000=12K
unsigned long m_DataStartAddress	数据段起始地址	0x00404000=4M+16K
unsigned long m_DataSize	数据段长度	0x00003000=12K
unsigned long m_StackSize	栈段长度	0x00001000=4K

- (3) 获取进程的 Proc 结构

Address	0 - 3	4 - 7	8 - B	C - F
C0119600	00000000	02000000	01000000	00F04000
C0119610	00500000	94AE11C0	03000000	01000000
C0119620	65000000	1A000000	00000000	00000000
C0119630	00000000	00000000	A00D12C0	00000000

变量名称	含义	值
short p_uid	用户 ID	0
int p_pid	进程标识数	2
int p_ppid	父进程标识数	1
unsigned long p_addr	user 结构即 ppda 区的物理地址	0x0040F000
unsigned int p_size	除共享正文段的长度, 以字节单位	0x00005000=20K
Text* p_textp	指向代码段 Text 结构的逻辑地址	0xC011AE94

ProcessState p_stat	进程调度状态	3=SRUN
int p_flag	进程标志位	1=SLOAD
int p_pri	进程优先数	65
int p_cpu	cpu 值, 用于计算 p_pri	19
int p_nice	进程优先数微调参数	0
int p_time	进程在盘上(内存内)驻留时间	0
unsigned long p_wchan	进程睡眠原因	0

(4) 获取进程代码段的 Text 结构

Monitors				
0xC011AE94 : 0xC011AE94 <Hex> New Renderings...				
0xC03FF000	Address	0 - 3	4 - 7	8 - B
0xC0119600	C011AE90	01000100	70460000	00C04000
0xC011AE94	C011AEA0	84EC11C0	01000100	00000000
	C011AEB0	00000000	00000000	00000000

变量名称	含义	值
int x_daddr	代码段在盘交换区上的地址	0x00004670
unsigned long x_caddr	代码段起始地址 (物理地址)	0x0040C000
unsigned int x_size	代码段长度, 以字节为单位	0x00003000 = 12K
Inode* x_iptr	内存 inode 地址	0xC011ECD0
Unsigned short x_count	共享正文段的进程数	1
unsigned long m_DataSize	共享该正文段且图像在内存的进程数	1
Unsigned short x_ccount	栈段长度	0x00001000=4K

(5) 进程图像完整信息

名称	逻辑地址	物理地址	大小
代码段	0x00401000	0x0040C000	12K
可交换部分	0xC03FF000	0x0040F000	20K
PPDA 区	0xC03FF000	0x0040F000	4K
数据段	0x00404000	0x00410000	12K
堆栈段		0x00413000	1K

(6) 获取代码段和可交换部分起始地址的方法

逻辑地址: 数据段和代码段: 从 proc 块中该进程的 u\_MemoryDescriptor 获取

User 结构: 固定的为 0xC03FF000

物理地址: 代码段: proc 结构->text 结构, 根据 x\_ccount 的值来判断物理地址是在内存的 x\_caddr 还是盘交换区的 x\_daddr

PPDA 区: proc 结构的 p\_addr 即可获得

### 三. 找到进程完整的页表

(1) 获取相对虚实地址映射表 (题目(2))

(只截取了部分)

0xC0119600	C0208FF0	04000000	04000000	04000000	04000000
0xC011AE94	C0209000	04000000	05000000	05100000	05200000
0xC0208000	C0209010	07100000	07200000	07300000	04000000
	C0209020	04000000	04000000	04000000	04000000
	C0209030	04000000	04000000	04000000	04000000

0xC0208000	C0209F80	04000000	04000000	04000000	04000000
	C0209F90	04000000	04000000	04000000	04000000
	C0209FA0	04000000	04000000	04000000	04000000
	C0209FB0	04000000	04000000	04000000	04000000
	C0209FC0	04000000	04000000	04000000	04000000
	C0209FD0	04000000	04000000	04000000	04000000
	C0209FE0	04000000	04000000	04000000	04000000
	C0209FF0	04000000	04000000	04000000	07400000

页号	地址	值	
		高 20 位页框号	低 12 位标志位(u/s r/w p)
0#	\	\	\
.....	\	\	\
1024#	\	\	\
1025#	0xC0208000~0xC0208003	0	005
1026#	0xC0209008~0xC020900B	1	005
1027#	0xC020900C~0xC020900F	2	005
1028#	0xC0209010~0xC0209013	1	007
1029#	0xC0209014~0xC0209017	2	007
1030#	0xC0209018~0xC020901B	3	007
1031#	0xC020901C~0xC020901F	0	004
.....	.....	.....	.....
2047#	0xC0209FFC~0xC0209FFF	4	007

## (2) 进程的物理页表 (题目(3))

物理页表包括 0x200 号, 0x201 号, 0x202 号, 0x203 号页表, 其逻辑地址为 0xc0200000, 0xc0201000, 0xc0202000, 0xc0203000

### ① 页目录

0xc0119600	Address	0 - 3	4 - 7
0xc011ae94	C0200000	27202000	27302000
0xc0208000	C0200010	00000000	00000000
0xc0200000	C0200020	00000000	00000000
	C0200030	00000000	00000000

0xc0200000	C02000F0	00000000	00000000	0
0xc0200000	C0200C00	23102000	00000000	0
	C0200C10	00000000	00000000	0

页号	地址	值	
		高 20 位页框号	低 12 位标志位(u/s r/w p)
0#	0xC0200000~0xC0200003	0x202	027 (课件上这里是不是写错了)
1#	0xC0200004~0xC0200007	0x203	027 (课件上这里是不是写错了)
.....	.....	.....	.....
768#	0xC0200C00~0xC0200C03	0x201	023
.....	.....	.....	.....

### ② 用户页表 1

- ◆ 0xc03ff000
- ◆ 0xc0119600
- ◆ 0xc011ae94
- ◆ 0xc0208000
- ◆ 0xc0200000
- ◆ 0xc0201000
- ◆ 0xc0202000

Address	0 - 3	4 - 7	8 - B	C - F
C0202000	67000000	04100000	04200000	04300000
C0202010	06400000	06500000	06600000	06700000
C0202020	06800000	06900000	06A00000	06B00000
C0202030	06C00000	06D00000	06E00000	06F00000
C0202040	06000100	06100100	06200100	06300100
C0202050	06400100	06500100	06600100	06700100
C0202060	06800100	06900100	06A00100	06B00100

页号	地址	值	
		高 20 位页框号	低 12 位标志位(u/s r/w p)
0#	0xC0202000~0xC0202003	\	\
1#	0xC0202004~0xC0202007	\	\
.....	.....	.....	.....
1023#	0xC0202FFC~0xC0202FFF	\	\

### ③用户页表 2

- ◆ 0xc03ff000
- ◆ 0xc0119600
- ◆ 0xc011ae94
- ◆ 0xc0208000
- ◆ 0xc0200000
- ◆ 0xc0201000
- ◆ 0xc0202000
- ◆ 0xc0203000

Address	0 - 3	4 - 7	8 - B	C - F
C0203000	06004000	65C04000	65D04000	65E04000
C0203010	67004100	67104100	67204100	66204100
C0203020	66304100	06904000	06A04000	06B04000
C0203030	06C04000	06D04000	06E04000	06F04000
C0203040	06004100	06104100	06204100	06304100
C0203050	06404100	06504100	06604100	06704100
C0203060	06804100	06904100	06A04100	06B04100
C0203FE0	007F8006	007F9006	007FA006	007FB006
C0203FF0	007FC006	007FD006	007FE006	00413067
C0204000	00000000	00000000	00000000	00000000

页号	地址	值	
		高 20 位页框号	低 12 位标志位(u/s r/w p)
0#	0xC0203000~0xC0203003	\	\
1#	0xC0203004~0xC0203007	0x40C	065
2#	0xC0203008~0xC020300B	0x40D	065
3#	0xC020300C~0xC020300F	0x40E	065
4#	0xC0203010~0xC0203013	0x410	067
5#	0xC0203014~0xC0203017	0x411	067
6#	0xC0203018~0xC020301B	0x412	067
.....	.....	.....	.....
1023#	0xC0203FFC~0xC0203FFF	0x413	067

### ④核心页表

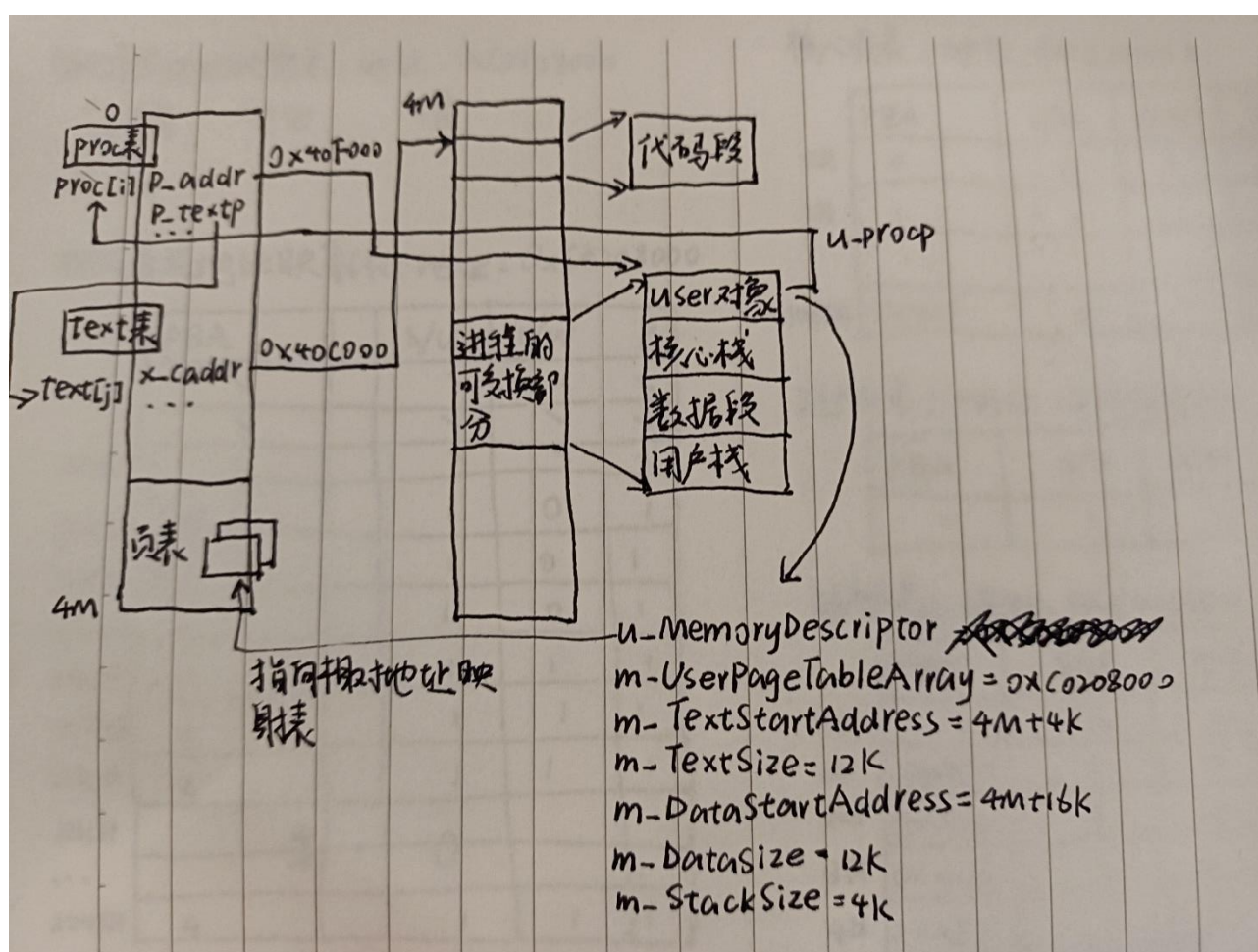
- ◆ 0xc03ff000
- ◆ 0xc0119600
- ◆ 0xc011ae94
- ◆ 0xc0208000
- ◆ 0xc0200000
- ◆ 0xc0201000

Address	0 - 3	4 - 7	8 - B	C - F
C0201000	03000000	03100000	03200000	03300000
C0201010	03400000	03500000	03600000	03700000
C0201020	03800000	03900000	03A00000	03B00000
C0201030	03C00000	03D00000	03E00000	23F00000
C0201040	03000100	03100100	03200100	03300100

0xc0200000	C0201FB0	003EC003	003ED003	003EE003	003EF003
0xc0201000	C0201FC0	003F0003	003F1003	003F2003	003F3003
0xc0202000	C0201FD0	003F4003	003F5003	003F6003	003F7003
0xc0203000	C0201FE0	003F8003	003F9003	003FA003	003FB003
	C0201FF0	003FC003	003FD003	003FE003	0040F063

页号	地址	值	
		高 20 位页框号	低 12 位标志位(u/s r/w p)
0#	0xC0201000~0xC0201003	0x001	003
1#	0xC0201004~0xC0201007	0x002	033
.....	.....	.....	.....
1023#	0xC0201FFC~0xC0201FFF	0x40F	063

#### 四. 完整的进程图像





相对虚拟地址映射表: 地址: 0xC0208000

	PBA		s/u	r/w	P
0#	\		\	\	\
...	\		\	\	\
1024#	\		\	\	\
1025#	0		1	0	1
1026#	1		1	0	1
1027#	2		1	0	1
1028#	1		1	1	1
1029#	2		1	1	1
1030#	3		1	1	1
1031#		全	0		
...					
2047#	4		1	1	1

核心页表: 地址: 0xC0201000

	PBA	s/u	r/w	P
0#	0	0	1	1
1#	1	0	1	1
...				
1023#	0x40F	0	1	1

用户页表1: 地址: 0xC0202000

PBA	s/u	r/w	P
\	\	\	\

用户页表2: 地址: 0xC0203000

	PBA	s/u	r/w	P
0#	\	\	\	\
1#	0x40C	1	0	1
2#	0x40D	1	0	1
3#	0x40E	1	0	1
4#	0x410	1	1	1
5#	0x411	1	1	1
6#	0x412	1	1	1
...				
1023#	0x413	1	1	1