

УНИВЕРСИТЕТ ИТМО
Факультет программной инженерии и компьютерной техники

Лабораторная работа 1

По дисциплине:
«Операционные системы»

г. Санкт-Петербург
2020

Лабораторная работа №1

Разработать программу на языке C, которая осуществляет следующие действия

- Создает область памяти размером 184 мегабайт, начинающихся с адреса 0x936A655 (если возможно) при помощи malloc заполненную случайными числами /dev/urandom в 11 потоков. Используя системные средства мониторинга определите адрес начала в адресном пространстве процесса и характеристики выделенных участков памяти. Замеры виртуальной/физической памяти необходимо снять:
 1. До аллокации
 2. После аллокации
 3. После заполнения участка данными
 4. После деаллокации
- Записывает область памяти в файлы одинакового размера 147 мегабайт с использованием block(блочного) обращения к диску. Размер блока ввода-вывода 115 байт. Преподаватель выдает в качестве задания последовательность записи/чтения блоков seq(последовательный)
- Генерацию данных и запись осуществлять в бесконечном цикле.
- В отдельных 114 потоках осуществлять чтение данных из файлов и подсчитывать агрегированные характеристики данных - avg(среднее значение).
- Чтение и запись данных в/из файла должна быть защищена примитивами синхронизации sema.
- По заданию преподавателя изменить приоритеты потоков и описать изменения в характеристиках программы.

Для запуска программы возможно использовать операционную систему Windows 10 или Debian/Ubuntu в виртуальном окружении.

Измерить значения затраченного процессорного времени на выполнение программы и на операции

ввода-вывода используя системные утилиты.

Отследить трассу системных вызовов.

Используя star построить графики системных характеристик.

<https://github.com/SunnyCapt/os/tree/master/l1>

Адрес начала в адресном пространстве и характеристики выделенных участков памяти: ***cat /proc/ps -C main | awk '{print \$1}' | tail -n 1`/maps*** или ***pmap -x `ps -C main | awk '{print \$1}' | tail -n 1`***

```
+ ~ sudo cat /proc/$pid/maps | head -n 10 && echo "..." && sudo cat /proc/$pid/maps | tail -n 50
55c801e72000-55c801e73000 r--p 00000000 fe:05 4465410 /home/fisox/projects/trash/os-itmo/ll/main
55c801e73000-55c801e74000 r-xp 00001000 fe:05 4465410 /home/fisox/projects/trash/os-itmo/ll/main
55c801e74000-55c801e75000 r--p 00002000 fe:05 4465410 /home/fisox/projects/trash/os-itmo/ll/main
55c801e75000-55c801e76000 r--p 00002000 fe:05 4465410 /home/fisox/projects/trash/os-itmo/ll/main
55c801e76000-55c801e77000 rw-p 00003000 fe:05 4465410 /home/fisox/projects/trash/os-itmo/ll/main
55c803184000-55c805323000 rw-p 00000000 00:00 0 [heap]
7fb070000000-7fb073230000 rw-p 00000000 00:00 0
7fb073230000-7fb074000000 ---p 00000000 00:00 0
7fb078000000-7fb07a176000 rw-p 00000000 00:00 0
7fb07a176000-7fb07c000000 ---p 00000000 00:00 0
...
7fb17a7fd000-7fb17a7fe000 ---p 00000000 00:00 0
7fb17a7fe000-7fb17affe000 rw-p 00000000 00:00 0
7fb17affe000-7fb17afff000 ---p 00000000 00:00 0
7fb17afff000-7fb17b7ff000 rw-p 00000000 00:00 0
7fb17b7ff000-7fb17b800000 ---p 00000000 00:00 0
7fb17b800000-7fb17c000000 rw-p 00000000 00:00 0
7fb17c000000-7fb17d0be000 rw-p 00000000 00:00 0
7fb17d0be000-7fb180000000 ---p 00000000 00:00 0
7fb180058000-7fb18005b000 r--p 00000000 fe:01 526532 /usr/lib/x86_64-linux-gnu/libgcc_s.so.1
7fb18005b000-7fb18006c000 r-xp 00003000 fe:01 526532 /usr/lib/x86_64-linux-gnu/libgcc_s.so.1
7fb18006c000-7fb18006f000 r--p 00014000 fe:01 526532 /usr/lib/x86_64-linux-gnu/libgcc_s.so.1
7fb18006f000-7fb180070000 ---p 00017000 fe:01 526532 /usr/lib/x86_64-linux-gnu/libgcc_s.so.1
7fb180070000-7fb180071000 r--p 00017000 fe:01 526532 /usr/lib/x86_64-linux-gnu/libgcc_s.so.1
7fb180071000-7fb180072000 rw-p 00018000 fe:01 526532 /usr/lib/x86_64-linux-gnu/libgcc_s.so.1
7fb180072000-7fb180073000 ---p 00000000 00:00 0
7fb180073000-7fb180073000 rw-p 00000000 00:00 0
7fb180073000-7fb180074000 ---p 00000000 00:00 0
7fb180074000-7fb181074000 rw-p 00000000 00:00 0
7fb181074000-7fb181075000 ---p 00000000 00:00 0
7fb181075000-7fb181875000 rw-p 00000000 00:00 0
7fb181875000-7fb181876000 ---p 00000000 00:00 0
7fb181876000-7fb18d87a000 rw-p 00000000 00:00 0
7fb18d87a000-7fb18d89c000 r--p 00000000 fe:01 526153 /usr/lib/x86_64-linux-gnu/libc-2.28.so
7fb18d89c000-7fb18d9e4000 r-xp 00022000 fe:01 526153 /usr/lib/x86_64-linux-gnu/libc-2.28.so
7fb18d9e4000-7fb18da30000 r--p 0016a000 fe:01 526153 /usr/lib/x86_64-linux-gnu/libc-2.28.so
7fb18da30000-7fb18da31000 ---p 001b6000 fe:01 526153 /usr/lib/x86_64-linux-gnu/libc-2.28.so
7fb18da31000-7fb18da35000 r--p 001b6000 fe:01 526153 /usr/lib/x86_64-linux-gnu/libc-2.28.so
7fb18da35000-7fb18da37000 rw-p 001ba000 fe:01 526153 /usr/lib/x86_64-linux-gnu/libc-2.28.so
7fb18da37000-7fb18da3b000 rw-p 00000000 00:00 0
7fb18da3b000-7fb18da41000 r--p 00000000 fe:01 527454 /usr/lib/x86_64-linux-gnu/libpthread-2.28.s
7fb18da41000-7fb18da50000 r-xp 00006000 fe:01 527454 /usr/lib/x86_64-linux-gnu/libpthread-2.28.s
7fb18da50000-7fb18da56000 r--p 00015000 fe:01 527454 /usr/lib/x86_64-linux-gnu/libpthread-2.28.s
7fb18da56000-7fb18da57000 r--p 0001a000 fe:01 527454 /usr/lib/x86_64-linux-gnu/libpthread-2.28.s
7fb18da57000-7fb18da58000 rw-p 0001b000 fe:01 527454 /usr/lib/x86_64-linux-gnu/libpthread-2.28.s
7fb18da58000-7fb18da5c000 rw-p 00000000 00:00 0
7fb18da5c000-7fb18da5e000 r--p 00000000 fe:01 527514 /usr/lib/x86_64-linux-gnu/librt-2.28.so
7fb18da5e000-7fb18da62000 r-xp 00002000 fe:01 527514 /usr/lib/x86_64-linux-gnu/librt-2.28.so
7fb18da62000-7fb18da64000 r--p 00006000 fe:01 527514 /usr/lib/x86_64-linux-gnu/librt-2.28.so
7fb18da64000-7fb18da65000 r--p 00007000 fe:01 527514 /usr/lib/x86_64-linux-gnu/librt-2.28.so
7fb18da65000-7fb18da66000 rw-p 00008000 fe:01 527514 /usr/lib/x86_64-linux-gnu/librt-2.28.so
7fb18da66000-7fb18da68000 rw-p 00000000 00:00 0
7fb18da83000-7fb18da84000 r--p 00000000 fe:01 525497 /usr/lib/x86_64-linux-gnu/ld-2.28.so
7fb18da84000-7fb18daa2000 r-xp 00001000 fe:01 525497 /usr/lib/x86_64-linux-gnu/ld-2.28.so
7fb18daa2000-7fb18daaa000 r--p 0001f000 fe:01 525497 /usr/lib/x86_64-linux-gnu/ld-2.28.so
7fb18daaa000-7fb18daab000 r--p 00026000 fe:01 525497 /usr/lib/x86_64-linux-gnu/ld-2.28.so
7fb18daab000-7fb18daac000 rw-p 00027000 fe:01 525497 /usr/lib/x86_64-linux-gnu/ld-2.28.so
7fb18daac000-7fb18daad000 rw-p 00000000 00:00 0
7ffd8ac3c000-7ffd8ac5d000 rw-p 00000000 00:00 0 [stack]
7ffd8acd6000-7ffd8acd9000 r--p 00000000 00:00 0 [vvar]
7ffd8acd9000-7ffd8acdb000 r-xp 00000000 00:00 0 [vdso]
```

Замеры виртуальной/физической памяти: ***ps -eo pid,vsz,rss,comm | grep main***

До аллокации

pid vsz rss

```
→ ~ ps -eo pid,vsz,rss,comm | grep main
25357 2460 740 main
```

После аллокации

pid vsz rss

```
→ ~ ps -eo pid,vsz,rss,comm | grep main
25357 190880 740 main
```

После заполнения данными

pid vsz rss

```
→ ~ ps -eo pid,vsz,rss,comm | grep main
25357 944664 190288 main
```

После деаллокации

pid vsz rss

```
→ ~ ps -eo pid,vsz,rss,comm | grep main
25357 756244 1868 main
```

Измерить значения затраченного процессорного времени на выполнение программы:

time ./main

```
./main 57.11s user 7.22s system 197% cpu 32.637 total
```

и на операции ввода-вывода используя системные утилиты: ***strace -c -f ./main***

% time	seconds	usecs/call	calls	errors	syscall
97.76	25505.317003	4248054	6004		futex
2.15	561.786727	34	16109929		read
0.07	19.267431	21	904029		write
0.01	1.390107	2242	620		openat
0.00	0.274953	143	1910		clone
0.00	0.172685	132	1306		munmap
0.00	0.158316	108	1455		mprotect
0.00	0.132582	99	1327		mmap
0.00	0.046958	24	1912		madvise
0.00	0.043525	22	1910		set_robust_list
0.00	0.011644	18	617		fstat
0.00	0.007160	12	596		close
0.00	0.001856	11	163	163	ioctl
0.00	0.000018	18	1		brk
100.00	26088.610965		17031779	163	total

Отследить трассу системных вызовов: **strace -f ./main**

```
+ ll git:(master) x sudo strace -f ./main
execve("./main", [ "./main" ], 0x7ffc46cc9e48 /* 16 vars */) = 0
brk(NULL)                               = 0x55d92970f000
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=107640, ...}) = 0
mmap(NULL, 107640, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f452cc90000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/librt.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\260#\0\0\0\0\0"... , 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=35808, ...}) = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f452cc8e000
mmap(NULL, 39904, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f452cc84000
mmap(0x7f452cc86000, 16384, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f452cc86000
mmap(0x7f452cc8a000, 8192, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x6000) = 0x7f452cc8a000
mmap(0x7f452cc8c000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x7000) = 0x7f452cc8c000
close(3)                                = 0
```

Используя stap построить графики системных характеристик: **stap -x `ps -C main | awk '{print \$1}' | tail -n 1` script**

```
+ scripts git:(master) x sudo stap -x `ps -C main | tail -n 1 | awk '{print $1}'` script
starting probe
^C
  name    opens    reads    MB tot    B avg    writes    MB tot    B avg
  main         2    934153    120      137    24285      89    4096
```

Вывод: узнал о новых для себя утилитах измерения производительности в линукс-системах, кроме того познал новые грани разработки на C.