

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение высшего  
образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**

по лабораторной работе «Управление памятью в ОС Linux»

Автор: Конюхов Ф.В

Факультет: ИТИП

Группа: М3211



**УНИВЕРСИТЕТ ИТМО**

Санкт-Петербург 2020

**Задачи работы:** проведите два виртуальных эксперимента в соответствии с требованиями и проанализируйте их результаты. Рекомендуется написать «следящие» скрипты и собирать данные, например, из вывода утилиты **top** автоматически с заданной периодичностью, например, 1 раз в секунду. Можно проводить эксперименты и фиксировать требуемые параметры и в ручном режиме, но в этом случае рекомендуется замедлить эксперимент, например, уменьшив размер добавляемой к массиву последовательности с 10 до 5 элементов.

**Ход работы:**

## Эксперимент №1

### 1 Часть:

Скрипт **mem.bash**

```
#!/bin/bash

a=0
b=0
declare -a array
rm report.log
touch report.log

while true
do
    let b++
    array+=(1 3 5 7 9 11 13 15 17 19)
    let a++
    if [[ $((a % 100000)) -eq 0 ]]
    then
        a=0
        echo ${#array[@]} >> report.log
    fi
done
```

Последняя запись журнала:

```
иментар 6010.0944341 [ 14881 1000 1488 55839 1 73728 264 0 1
зданн [ 6010.0948121 [ 20401 1000 2040 83750 0 270336 225 0 su
ообщ [ 6010.0951821 [ 20441 0 2044 59090 1 86016 462 0 bash
с кото [ 6010.0956601 [ 21511 0 2151 664703 422669 4968448 186472 0 mem.bash
a rep [ 6010.0960431 Out of memory: Killed process 2151 (mem.bash) total-vm:2658812kB, anon-rss:1690676kB,
file-rss:0kB, shmem-rss:0kB, UID:0
: две к [ 6010.2223821 oom_reaper: reaped process 2151 (mem.bash), now anon-rss:0kB, file-rss:0kB, shmem-rss
: соль. U killed
их изм [root@localhost lab5]#
```

Последняя строка файла report.log:

```
е две к 26000000
соль. U 27000000
их изм 28000000
ения п 29000000
30000000
31000000
```

Следящий скрипт **check**:

```
# !bin/bash

./mem.bash&pid=$!
echo -e "Pid of mem.bash $pid\n" > fres
rm fres
rm fpid
while true
do
    top5pids=$(top -b -n 1 | head -12 | tail -n +8 | awk '{print ""$1""}' | paste -s -d " ")
    curPid=$(top -p $pid -b -n 1 | head -8 | tail -n +8)
    curPid=$(echo -e $curPid | awk '{print ""$11" "$10" "$5" "$6" "$7" "$9" "}"')
    freeMem=$(top -b -n 1 | head -4 | tail -n +4 | awk '{print ""$6""}')
    s=$(top -b -n 1 | head -5 | tail -n +5 | awk '{print ""$5""}')
    if ps -p $pid > /dev/null
    then
        res=$(echo -e "$curPid $freeMem $s\n")
        echo $res >> fres
        echo $top5pids >> fpid
    fi
done
```

Данные из эксперимента, полученные с помощью утилиты **top**

**5 топовых pid:**

```
check fpid fres mem.bash report.log
[root@localhost lab5]# cat fpid
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
3742 1 2 3 4
```

**freeMem и Swap**

```
[root@localhost lab5]# cat fres
607.3 730.3
576.0 730.3
545.2 730.3
514.3 730.3
483.5 730.3
452.2 730.3
421.1 730.3
391.5 730.3
360.1 730.3
328.8 730.3
298.3 730.3
267.1 730.3
236.2 730.3
```

Мы видим что графа со свободной памятью стремительно сокращается при исполнении бесконечного цикла.

Последние две записи в системном журнале (`dmesg | grep "mem.bash"`):

```
[root@localhost lab5]# dmesg | grep "mem.bash"
[ 3848.837582] [ 1577]      0 1577  664241  422227  4952064  186458          0 mem.bash
[ 3848.839730] Out of memory: Killed process 1577 (mem.bash) total-vm:2656964kB, anon-rss:1688900kB,
file-rss:0kB, shmem-rss:0kB, UID:0
[root@localhost lab5]#
```

Последняя строка в report.log:

```
3000000
4000000
5000000
```

## 2 Часть:

Скрипт `inner`

```
# !/bin/bash
!/.check&pid1=$!
!/.check2&pid=$!_
/m
```

Top 5 pid:

Результаты `fpid` и `fpid2`

```
4178 4176 1 2 3
4176 4178 1 2 3
4176 4178 1 2 3
4176 4178 47 1 2
4178 4176 1 2 3
4176 4178 4175 1 2
4178 4176 47 1 2
4176 4178 1 2 3
4178 4176 1 2 3
4176 4178 1 2 3
4176 4178 1 2 3
4178 4176 47 9 1
4176 4178 4175 1 2
4176 4178 47 1 2
4178 4176 1 2 3
4176 4178 47 1 2
4178 4176 1 2 3
4176 4178 1 2 3
4178 4176 1 2 3
4176 4178 1 2 3
4178 6887 1 2 3
```

freeMem и Swap:

Результаты fres и fres2

	B	234.4	159.8
		208.8	159.8
		180.7	159.8
		151.7	159.8
more	or	126.9	159.8
0kB	kB	101.4	159.8
er:	:	77.3	159.8
		52.8	154.3
		76.7	112.8
		66.5	84.2
memory	or	71.9	62.4
s:0	:0	70.1	41.1
er:	:	64.3	5.7

При использовании 2-х скриптов одновременно кроме стремительного сокращения памяти также быстрее активируется **Swap**

Последние две записи в системном журнале (`dmesg | grep "mem[2]*.bash"`):

```
31000000
[root@localhost lab5]# dmesg | grep "mem[2]*.bash"
[ 224.029696] [ 1500]    0 1500 358562 214430 2514944 88521          0 mem.bash
[ 224.030079] [ 1501]    0 1501 362753 208965 2539520 98190          0 mem.bash
[ 224.030468] Out of memory: Killed process 1501 (mem.bash) total-vm:1451012kB, anon-rss:835860kB,
file-rss:0kB, shmem-rss:0kB, UID:0
[ 257.461688] [ 1500]    0 1500 664076 421769 4964352 186757          0 mem.bash
[ 257.462093] Out of memory: Killed process 1500 (mem.bash) total-vm:2656304kB, anon-rss:1687076kB,
file-rss:0kB, shmem-rss:0kB, UID:0
[ 257.571805] oom_reaper: reaped process 1500 (mem.bash), now anon-rss:0kB, file-rss:0kB, shmem-rss:
:0kB
[root@localhost lab5]#
```

Последняя строка в report.log:

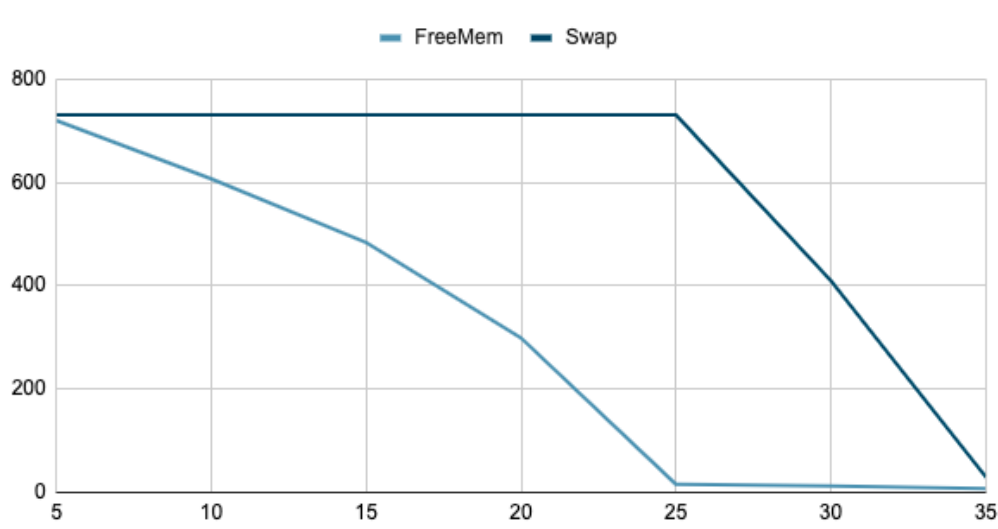
```
30000000
31000000
[root@localhost lab5]#
```

Последняя строка в report2.log:

```
30000000
31000000
```

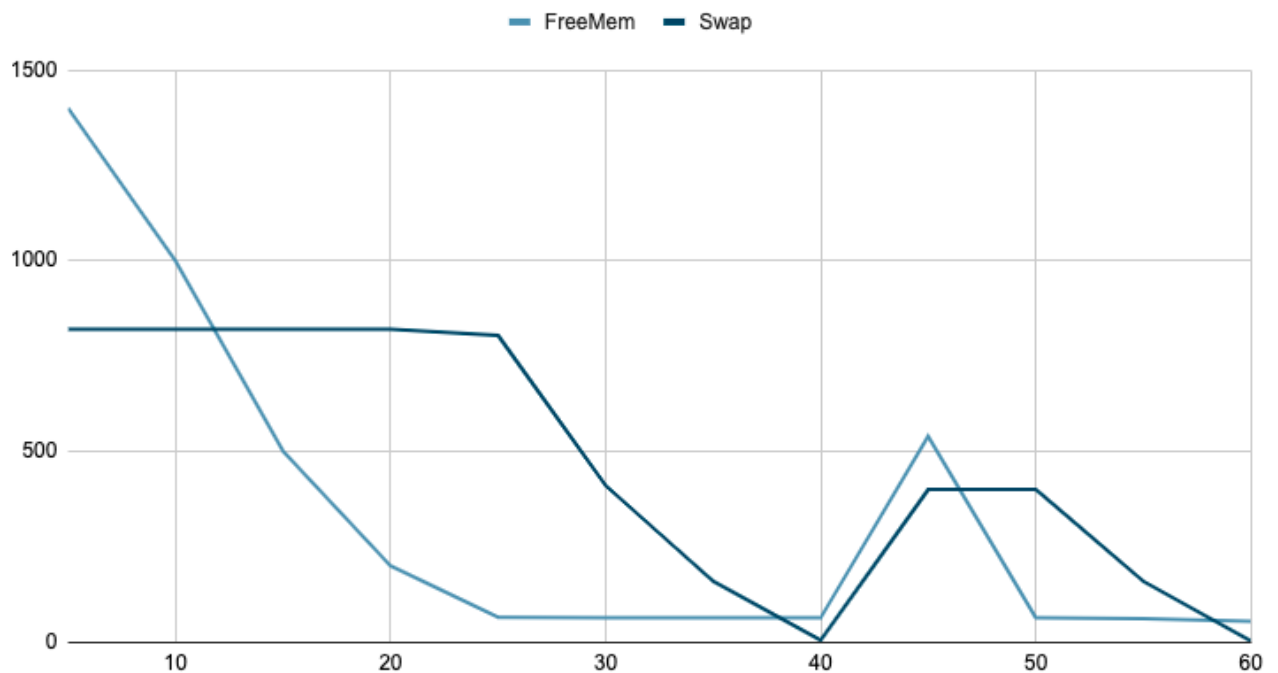
## Графики

### 1 Часть:



Сначала скрипт сокращал количество свободной памяти **freeMem**, а затем активировался раздел **Swap**, который и привел нас к аварийному завершению программы

### 2 Часть:



Сначала скрипт сокращал количество свободной памяти **freeMem**, а затем активировался раздел **Swap**, который и привел нас к аварийному завершению программы **mem.bash**, после 38 секунды активируется взлет **freeMem** и **Swap**, ведь завершение первого скрипта дало память второму **mem2.bash**

Аварийная остановка при  $N = 31000000$

## Эксперимент №2

Скрипт newmem.bash

```
#!/bin/bash
declare -a array

if [[ $# == 0 ]]; then
    N=31000000
else
    N=$1
fi

while true
do
    array+=(1 3 5 7 9 11 13 15 17 19)
    if (( "${#array[@]}" >= "$N" ))
    then
        echo "Ok, yeet"
        exit
    fi
done

[root@localhost lab5]# ./newmem.bash 19000000
Ok, yeet
```

## Скрипт **maincheck**

```
#!/bin/bash

if [[ $# == 0 ]]; then
    N=31000000
    K=10
else
    N=$1
    K=$2
fi

for (( i=0; i < $K; i++ ))
do
    ./newmem.bash $N&
    sleep 1s
done
```

При N в 10 раз меньше критического и K=10 запускам все процессы завершаются корректно

```
[root@localhost lab5]# ./newmem.bash 3100000
Ok, yeet
[root@localhost lab5]# ./maincheck 3100000 10
[root@localhost lab5]# Ok, yeet

[root@localhost lab5]# Ok, yeet
Ok, yeet
Ok, yeet
Ok, yeet
Ok, yeet
Ok, yeet
Ok, yeet
Ok, yeet
Ok, yeet

[root@localhost lab5]# _
```

В системном журнале ошибок не обнаружено!



[illegible][illegible][illegible]

При **N=2100000** , а **K=30** , только **2** процесса завершились аварийно!

[illegible]

При  $N=2000000$ , а  $K=30$ , все завершилось корректно!

[illegible]

**Ответ: N = 2000**