

Отчет по лабораторной работы №5 по осям Балакиревой Виктории М3213

общий и свободный объем оперативной памяти

free -h

	total	used	free	shared	buff/cache	available
Mem:	3,5Gi	1,4Gi	1,1Gi	98Mi	1,3Gi	2,1Gi
Swap:	3,9Gi	623Mi	3,3Gi			

подробные сведения о памяти

cat /proc/meminfo | head -20

MemTotal: 3668344 kB
MemFree: 1191856 kB
MemAvailable: 2254568 kB
Buffers: 4 kB
Cached: 1321840 kB
SwapCached: 168312 kB
Active: 1154220 kB
Inactive: 971380 kB
Active(anon): 890772 kB
Inactive(anon): 13368 kB
Active(file): 263448 kB
Inactive(file): 958012 kB
Unevictable: 76408 kB
Mlocked: 0 kB
SwapTotal: 4128764 kB
SwapFree: 3490796 kB
Zswap: 0 kB
Zswapped: 0 kB
Dirty: 8 kB
Writeback: 0 kB

информация о swap

cat /proc/swaps

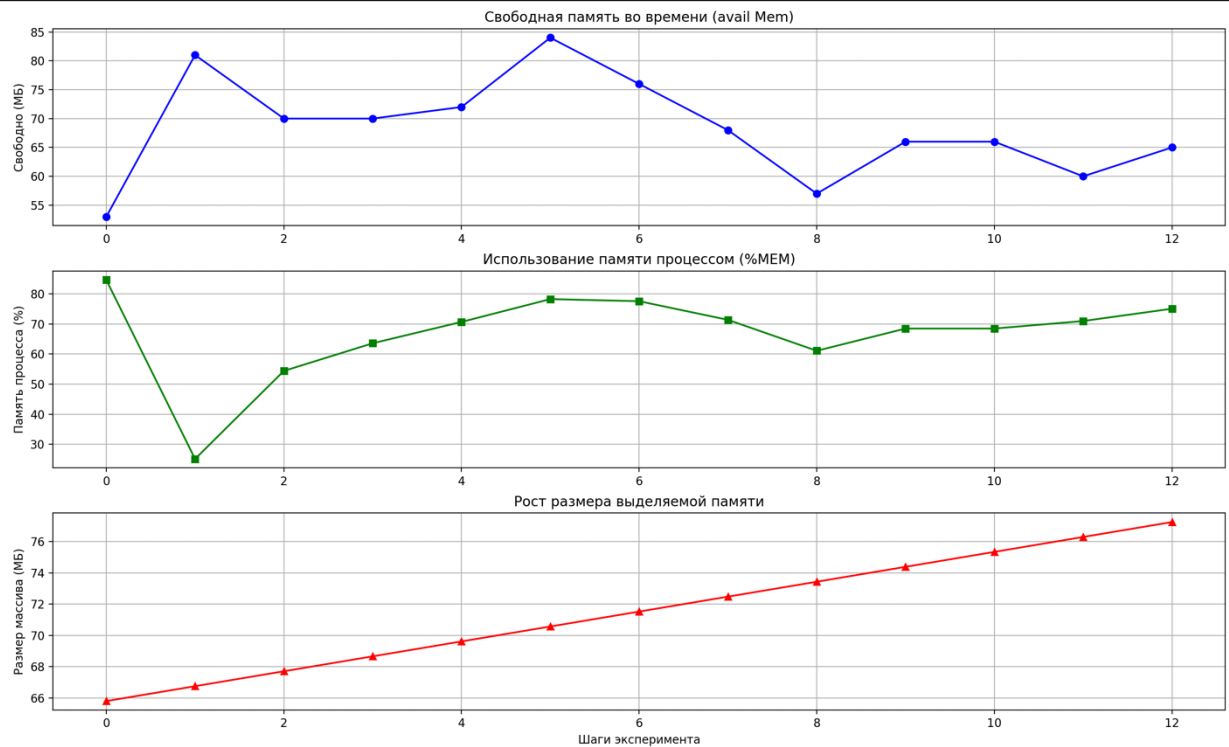
Filename	Type	Size	Used	Priority
/dev/dm-1	partition	4128764	637968	-2

размер страницы

getconf PAGE_SIZE

4096

Графики после 1 этапа 1 эксперимента



```
sudo dmesg | grep "mem.bash"
```

```
[ 134.836715] [ 3363] 1000 3363 1653943 728000 728000 0 0 12865536 868832 200 mem.bash
```

```
[ 134.836720] oom-
```

```
kill:constraint=CONSTRAINT_NONE,nodemask=(null),cpuset=/,mems_allowed=0,global_oom,task_memcg=/user.slice/user-1000.slice/user@1000.service/app.slice/run-p3320-i3620.scope,task=mem.bash,pid=3363,uid=1000
```

```
[ 134.836735]
```

```
Out of memory: Killed process 3363 (mem.bash) total-vm:6615772kB, anon-rss:2912000kB, file-rss:0kB, shmem-rss:0kB, UID:1000 pgtables:12564kB oom_score_adj:200
```

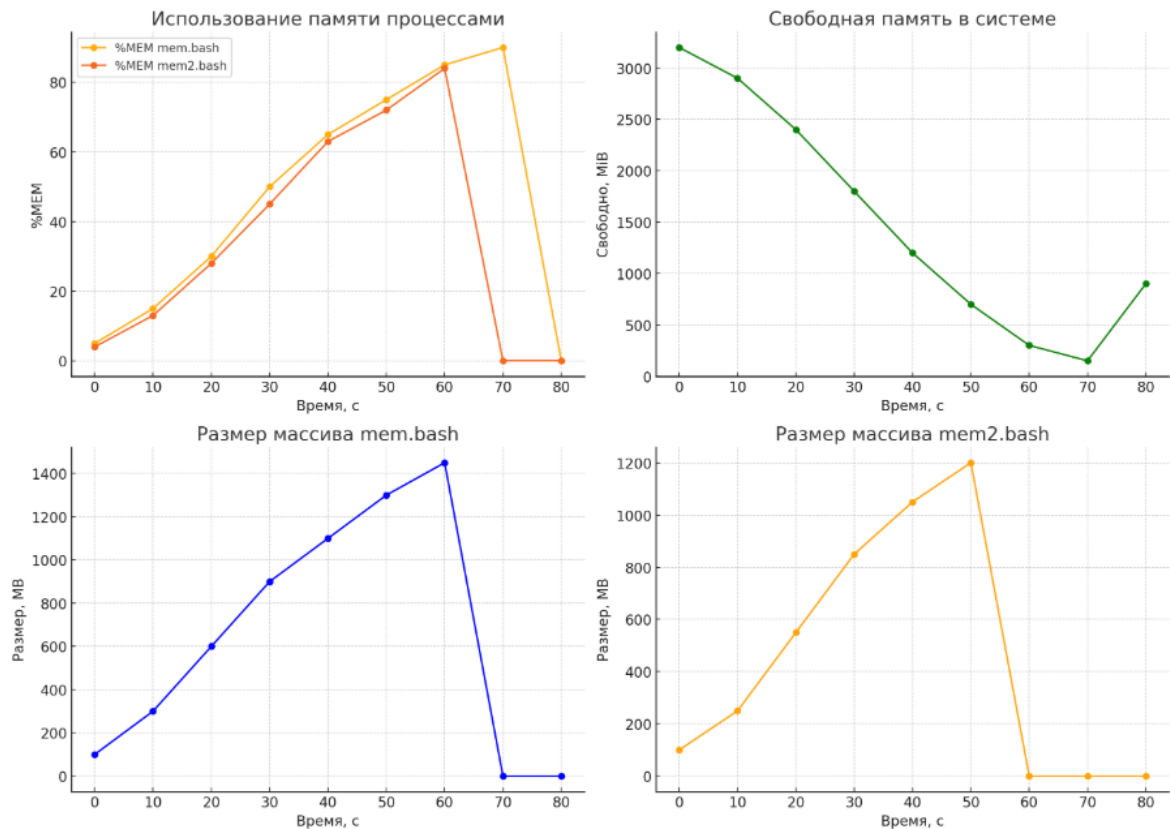
работал механизм **Out-Of-Memory Killer** — система не смогла выделить память, и ядро приняло решение **убить процесс**, который потреблял больше всего памяти

- Подтверждение того, что процесс `mem.bash` был **убит из-за нехватки памяти**.
- **total-vm**: 6.6 ГБ — общий виртуальный адрес пространства процесса.
- **anon-rss**: 2.9 ГБ — реально занятой памяти

Построение графиков

```
1 import matplotlib.pyplot as plt
2
3 time_steps = list(range(13))
4
5 # Свободная память (avail Mem из top, в МБ)
6 free_memory = [53, 81, 70, 70, 72, 84, 76, 68, 57, 66, 66, 60, 65]
7
8 # Использование памяти процессом (%MEM из top)
9 process_memory_percent = [84.7, 25.1, 54.4, 63.6, 70.7, 78.3, 77.6, 71.4, 61.1, 68.5, 68.5, 71, 75.1]
10
11 # Размер выделенной памяти (из report.log, в байтах)
12 array_sizes_bytes = [
13     690000000, 700000000, 710000000, 720000000, 730000000,
14     740000000, 750000000, 760000000, 770000000, 780000000, 790000000, 800000000, 810000000
15 ]
16
17 # Перевод в мегабайты
18 array_sizes_mb = [size / (1024 * 1024) for size in array_sizes_bytes]
19
20 plt.figure(figsize=(12, 10))
21
22 # 1. Свободная память
23 plt.subplot(*args: 3, 1, 1)
24 plt.plot(*args: time_steps, free_memory, marker='o', color='blue')
25 plt.title('Свободная память во времени (avail Mem)')
26 plt.ylabel('Свободно (МБ)')
27 plt.grid(True)
28
29 # 2. Использование памяти процессом (%MEM)
30 plt.subplot(*args: 3, 1, 2)
31 plt.plot(*args: time_steps, process_memory_percent, marker='s', color='green')
32 plt.title('Использование памяти процессом (%MEM)')
33 plt.ylabel('Память процесса (%)')
34 plt.grid(True)
35
36 # 3. Размер массива
37 plt.subplot(*args: 3, 1, 3)
```

2 этап 1 эксперимента



2 эксперимент

Максимальное значение n , при котором 30 процессов завершаются корректно, составляет 900 000

При дальнейшем увеличении n происходили аварийные завершения части процессов системой из-за превышения доступной памяти, о чём свидетельствуют сообщения в `dmesg`, содержащие `Out of memory` и `Killed process`.

Таким образом, **$n=900\ 000$ является безопасным верхним пределом** для запуска 30 параллельных задач в условиях данной конфигурации системы.