

Тестирование программного обеспечения

Лабораторная работа 8

Профилирование ПО

Дорогин Никита ИП-312

Цель работы

Освоение методов профилирования ПО для выявления узких мест в производительности.

Профилирование QEMU

Настройка Perf:

```
gastello123@gastello123:~$ perf --version
WARNING: perf not found for kernel 6.14.0-29

You may need to install the following packages for this specific kernel:
  linux-tools-6.14.0-29-generic
  linux-cloud-tools-6.14.0-29-generic

You may also want to install one of the following packages to keep up to date:
  linux-tools-generic
  linux-cloud-tools-generic
gastello123@gastello123:~$
```

```
gastello123@gastello123:~$ sudo nano .bashrc
```

```
alias Perf='/usr/lib/linux-tools/*/perf'
```

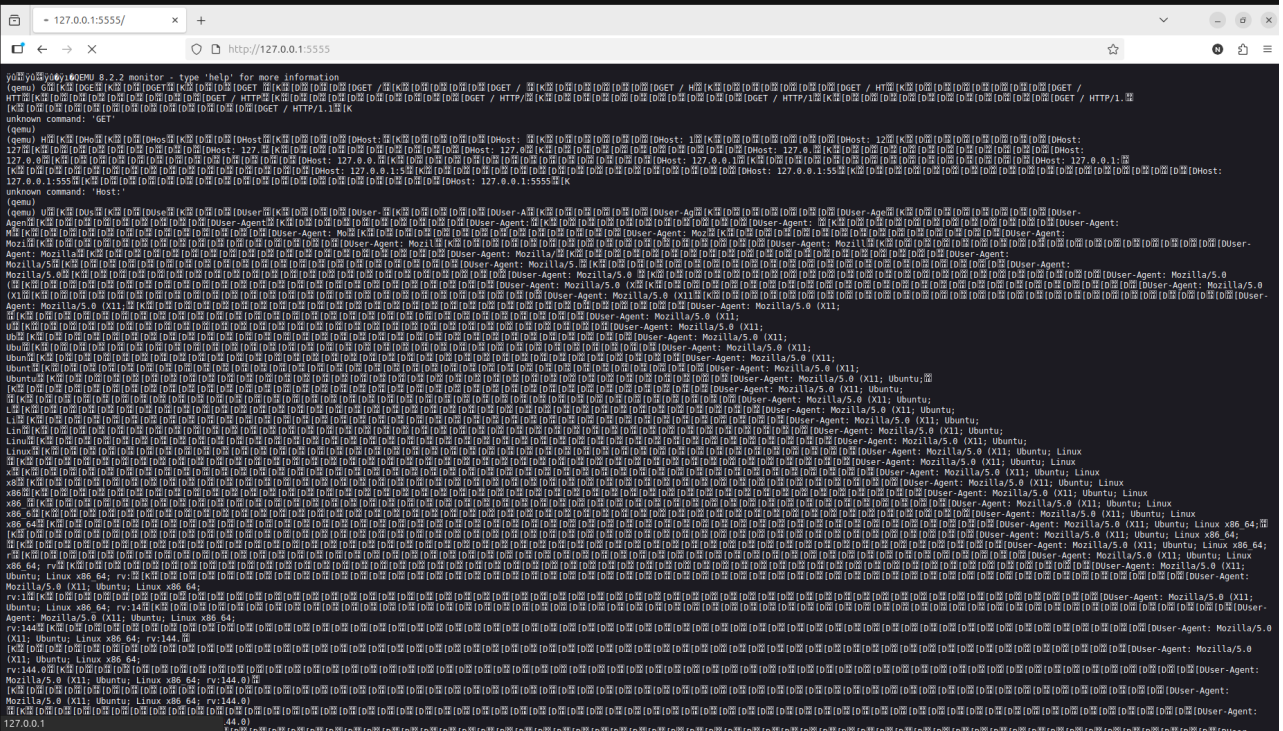
```
gastello123@gastello123:~$ Perf stat -e cycles,instructions,cache-misses qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio
Error:
Access to performance monitoring and observability operations is limited.
Consider adjusting /proc/sys/kernel/perf_event_paranoid setting to open
access to performance monitoring and observability operations for processes
without CAP_PERFMON, CAP_SYS_PTRACE or CAP_SYS_ADMIN Linux capability.
More information can be found at 'Perf events and tool security' document:
https://www.kernel.org/doc/html/latest/admin-guide/perf-security.html
perf_event_paranoid setting is 4:
-1: Allow use of (almost) all events by all users
    Ignore mlock limit after perf_event_mlock_kb without CAP_IPC_LOCK
>= 0: Disallow raw and ftrace function tracepoint access
>= 1: Disallow CPU event access
>= 2: Disallow kernel profiling
To make the adjusted perf_event_paranoid setting permanent preserve it
in /etc/sysctl.conf (e.g. kernel.perf_event_paranoid = <setting>)
gastello123@gastello123:~$ echo 'kernel.perf_event_paranoid = 1' | sudo tee -a /etc/sysctl.conf
kernel.perf_event_paranoid = 1
gastello123@gastello123:~$ sudo sysctl -p
kernel.perf_event_paranoid = 1
gastello123@gastello123:~$
```

Сбор метрик QEMU:

```

: Allow use of (almost) all events by all users
: Ignore mlock limit after perf_event_mlock_kb without CAP_IPC_LOCK
: Disallow raw and ftrace function tracepoint access
: Disallow CPU event access
: Disallow kernel profiling
ake the adjusted perf_event_paranoid setting permanent
etc/sysctl.conf (e.g. kernel.perf_event_paranoid = <settings>)
llo123@gastello123:~$ sudo Perf stat -e cycles,inst
: пароль для gastello123:
: Perf: команда не найдена
llo123@gastello123:~$ Perf stat -e cycles,instructions,cache-misses qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio
:
ss to performance monitoring and observability
ider adjusting /proc/sys/kernel/perf_event_paranoid to 1
ss to performance monitoring and observability
out CAP_PERFMON, CAP_SYS_PTRACE or CAP_SYS_ADMIN
Information can be found at 'Perf events and tracing'
s://www.kernel.org/doc/html/latest/admin-guide/proc/sys/kernel/perf_event_paranoid.html
event_paranoid setting is 4:
: Allow use of (almost) all events by all users
: Ignore mlock limit after perf_event_mlock_kb without CAP_IPC_LOCK
: Disallow raw and ftrace function tracepoint access
: Disallow CPU event access
: Disallow kernel profiling
ake the adjusted perf_event_paranoid setting permanent
etc/sysctl.conf (e.g. kernel.perf_event_paranoid = <settings>)
llo123@gastello123:~$ sudo sysctl -p
el.perf_event_paranoid = 1
llo123@gastello123:~$ sudo sysctl -p
el.perf_event_paranoid = 1
llo123@gastello123:~$ Perf stat -e cycles,instructions,cache-misses qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio
:

```



Выключаем QEMU

```

gastello123@gastello123:~$ Perf stat -e cycles,instructions,cache-misses qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio

Performance counter stats for 'qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio':

57919479997      cycles
149864780781     instructions          #    2.59  insn per cycle
46488759         cache-misses

22.186157609 seconds time elapsed

13.373836000 seconds user
0.293985000 seconds sys

```

Теперь уже включаем с целью посмотреть информацию о памяти:

```
gastello123@gastello123:~$ telnet 127.0.0.1 5555
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
QEMU 8.2.2 monitor - type 'help' for more information
(qemu) info mem
```

```
(qemu) info memdev
memory backend: ram
  size: 536870912
  merge: true
  dump: true
  prealloc: false
  share: false
  reserve: true
  policy: default
  host nodes:

(qemu) info memory-devices
(qemu) info memory_size_summary
base memory: 536870912
plugged memory: 0
(qemu) █
```

Сбор метрик хоста с помощью vmstat:

С выключенным QEMU

```
gastello123@gastello123:~$ vmstat 1
procs -----memory----- --swap-- -----io----- -system-- -----cpu-----
 r b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st gu
 1 0     0 8243372 93668 2636260    0    0   1531   259 3524    2  3  1 96  0  0  0
 1 0     0 8242416 93668 2636268    0    0     0     0 7409 16698  7  1 92  0  0  0
 3 0     0 8242416 93668 2636268    0    0     0     0 9392 20773  7  1 92  0  0  0
 1 0     0 8233944 93668 2636268    0    0     0     4 8089 17047  7  1 92  0  0  0
 1 0     0 8165648 93668 2639168    0    0     0     0 10446 19305  8  2 90  0  0  0
 1 0     0 8228400 93680 2639152    0    0     0    424 7878 17521  7  1 92  0  0  0
 1 0     0 8228400 93680 2639152    0    0     0    824 7154 16647  7  1 92  0  0  0
 1 0     0 8228652 93680 2639152    0    0     0     0 6954 16750  7  1 92  0  0  0
 1 0     0 8228652 93680 2639152    0    0     0     4 7215 17547  7  1 92  0  0  0
 1 0     0 8229408 93680 2639152    0    0     0     0 7108 17207  7  1 92  0  0  0
 1 0     0 8229408 93680 2639152    0    0     0     0 6256 14963  7  1 92  0  0  0
 1 0     0 8229408 93716 2639152    0    0     0    324 7723 19079  7  1 92  0  0  0
 1 0     0 8227644 93724 2639152    0    0     0    16 6419 14390  7  1 92  0  0  0
 1 0     0 8227644 93724 2639152    0    0     0     8 6920 16497  7  1 92  0  0  0
 1 0     0 8227644 93724 2639152    0    0     0     0 7417 17297  7  1 92  0  0  0
 1 0     0 8227620 93724 2639152    0    0     0    312 7361 17295  7  1 92  0  0  0
 1 0     0 8231844 93724 2639176    0    0     0     0 6723 17984  7  0 93  0  0  0
 1 0     0 8226104 93732 2639192    0    0     0    72 6631 16301  7  1 92  0  0  0
 1 0     0 8226104 93732 2639192    0    0     0   120 7288 19208  7  1 92  0  0  0
 1 0     0 8226104 93732 2639192    0    0     0     0 7333 16923  7  1 92  0  0  0
 3 0     0 8226104 93732 2639192    0    0     0     0 6848 16441  7  1 92  0  0  0
procs -----memory----- --swap-- -----io----- -system-- -----cpu-----
 r b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st gu
 1 0     0 8226936 93732 2639192    0    0     0   172 7142 18173  7  0 93  0  0  0
^C
```

С включенным QEMU

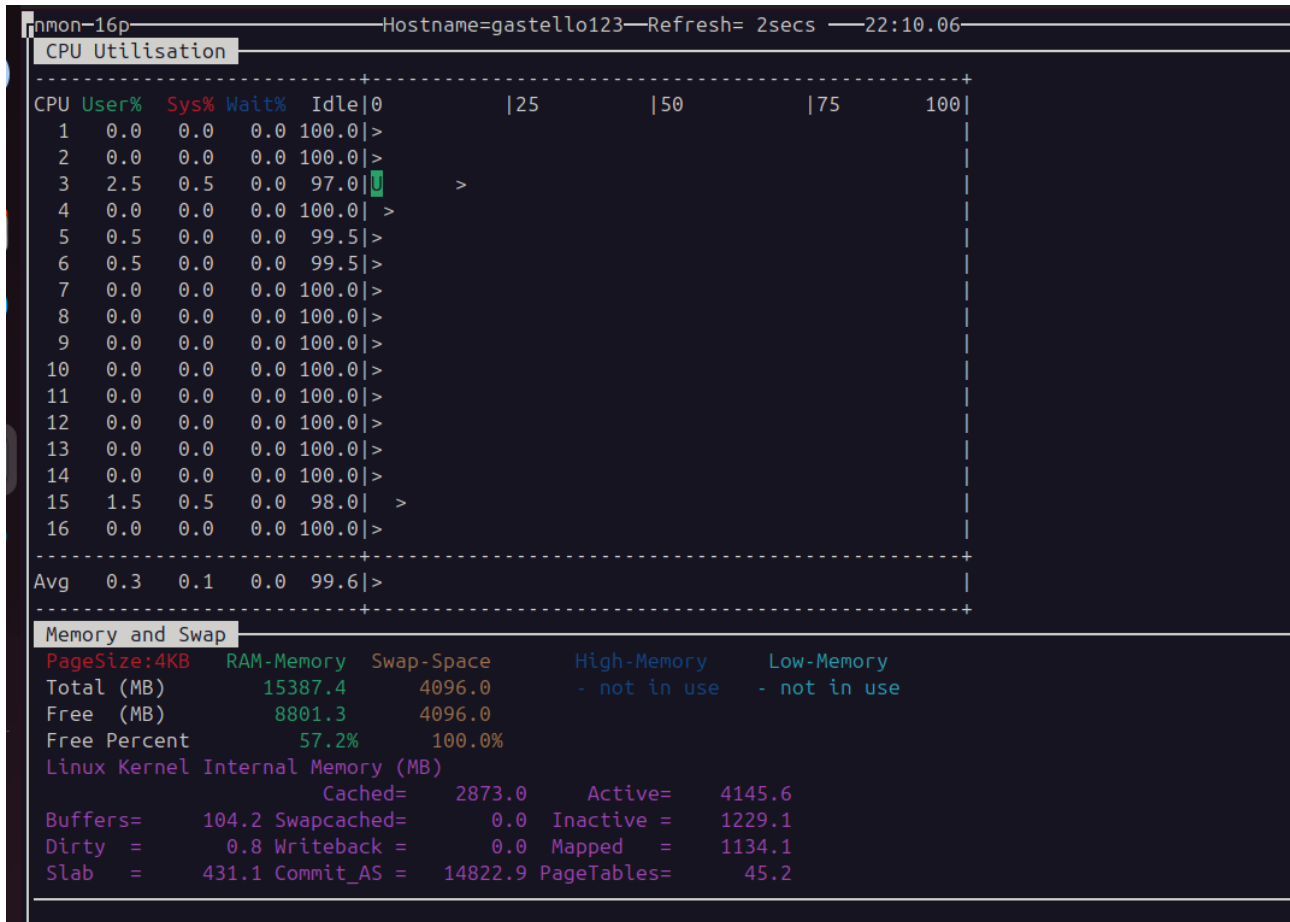
```
gastello123@gastello123:~$ vmstat 1
procs -----memory----- ---swap-- -----io---- -system-- -----cpu-----
 r  b    swpd    free    buff  cache   si   so    bi    bo    in  cs  us  sy  id  wa  st  gu
 1  0      0 9621840  87832 2477936    0    0   2126   256 2686    1  1  1  98  0  0  0
 1  0      0 9621864  87832 2478000    0    0    0    0 2756 1845  6  0  93  0  0  0
 1  0      0 9625116  87840 2478096    0    0    0   32 2712 1830  6  0  93  0  0  0
 1  0      0 9625116  87840 2478096    0    0    0    0 3583 3082  6  0  93  0  0  0
 1  0      0 9604004  87840 2483560    0    0    0    0 4627 3507  7  0  93  0  0  0
 1  0      0 9615244  87840 2483560    0    0    0    0 2605 1928  6  0  93  0  0  0
 1  0      0 9613480  87840 2483560    0    0    0    0 1491  375  6  0  94  0  0  0
 3  0      0 9613228  87848 2483560    0    0    0   52 1402  485  6  0  94  0  0  0
 1  0      0 9631788  87848 2483560    0    0    0    0 1661  523  6  0  94  0  0  0
 1  0      0 9622020  87848 2483560    0    0    0  392 2111 1464  6  0  94  0  0  0
 1  0      0 9626976  87876 2483560    0    0    0  344 1599  543  6  0  94  0  0  0
 1  0      0 9624120  87876 2483560    0    0    0    0 1466  464  6  0  94  0  0  0
 2  0      0 9624120  87876 2483560    0    0    0    0 1381  304  6  0  94  0  0  0
 1  0      0 9632416  87876 2483560    0    0    0    0 1587  466  7  0  93  0  0  0
 1  0      0 9625848  87876 2483624    0    0   128    0 1467  411  6  0  94  0  0  0
 1  0      0 9625440  87884 2483752    0    0    0  276 1422  388  6  0  94  0  0  0
 1  0      0 9623960  87884 2483756    0    0    0  576 1769  606  6  0  93  0  0  0
 1  0      0 9624680  87884 2483756    0    0    0    0 1406  395  6  0  94  0  0  0
 1  0      0 9633896  87884 2483756    0    0    0    0 1464  396  6  0  94  0  0  0
^C
gastello123@gastello123:~$
```

Сбор сетевых метрик с помощью nmon:

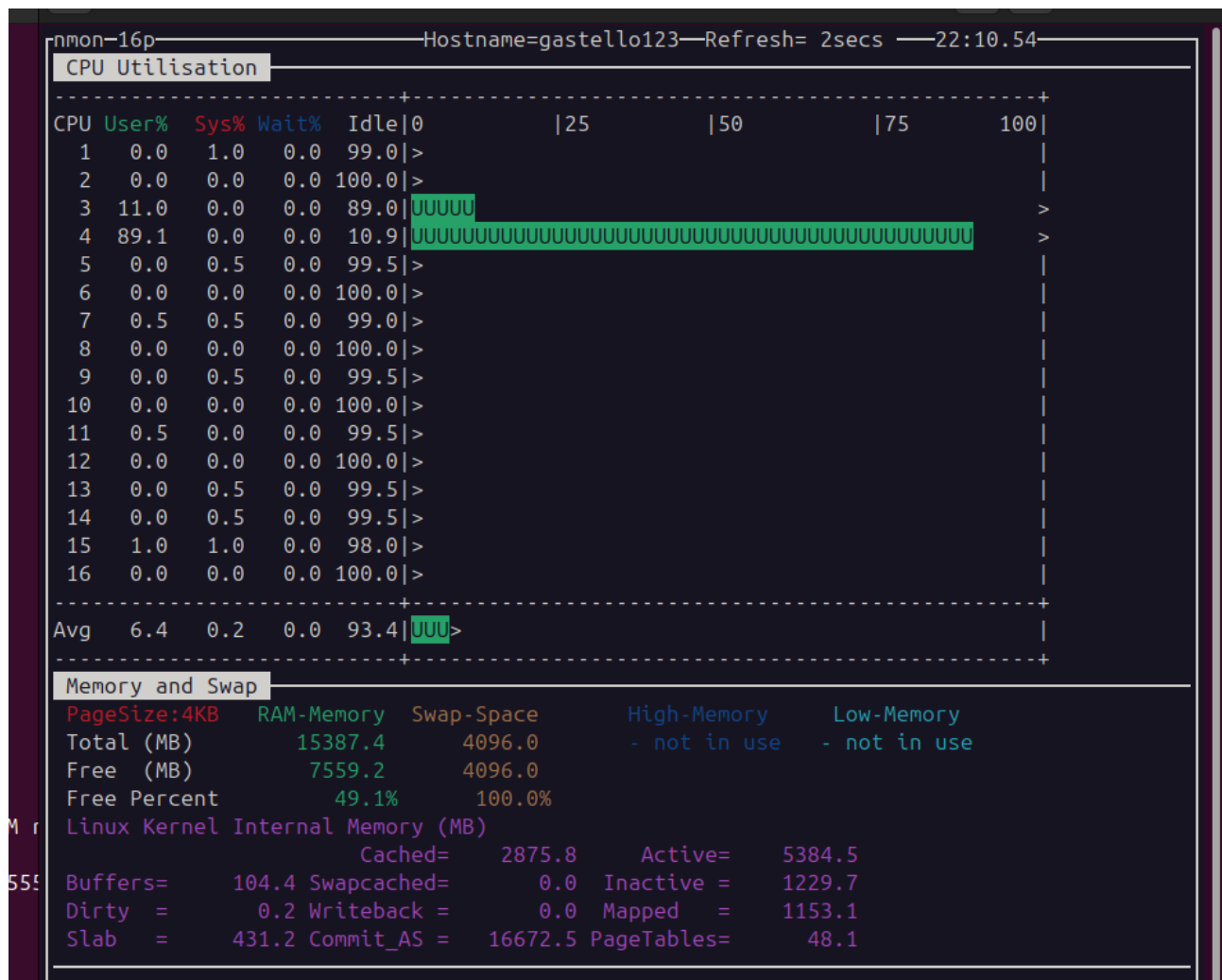
```
mon-16p-----Hostname=gastello123---Refresh= 2secs ---21:30.05-----
  nmon
  -----
  For help type H or ...
  nmon -? - hint
  nmon -h - full details
  To stop nmon type q to Quit

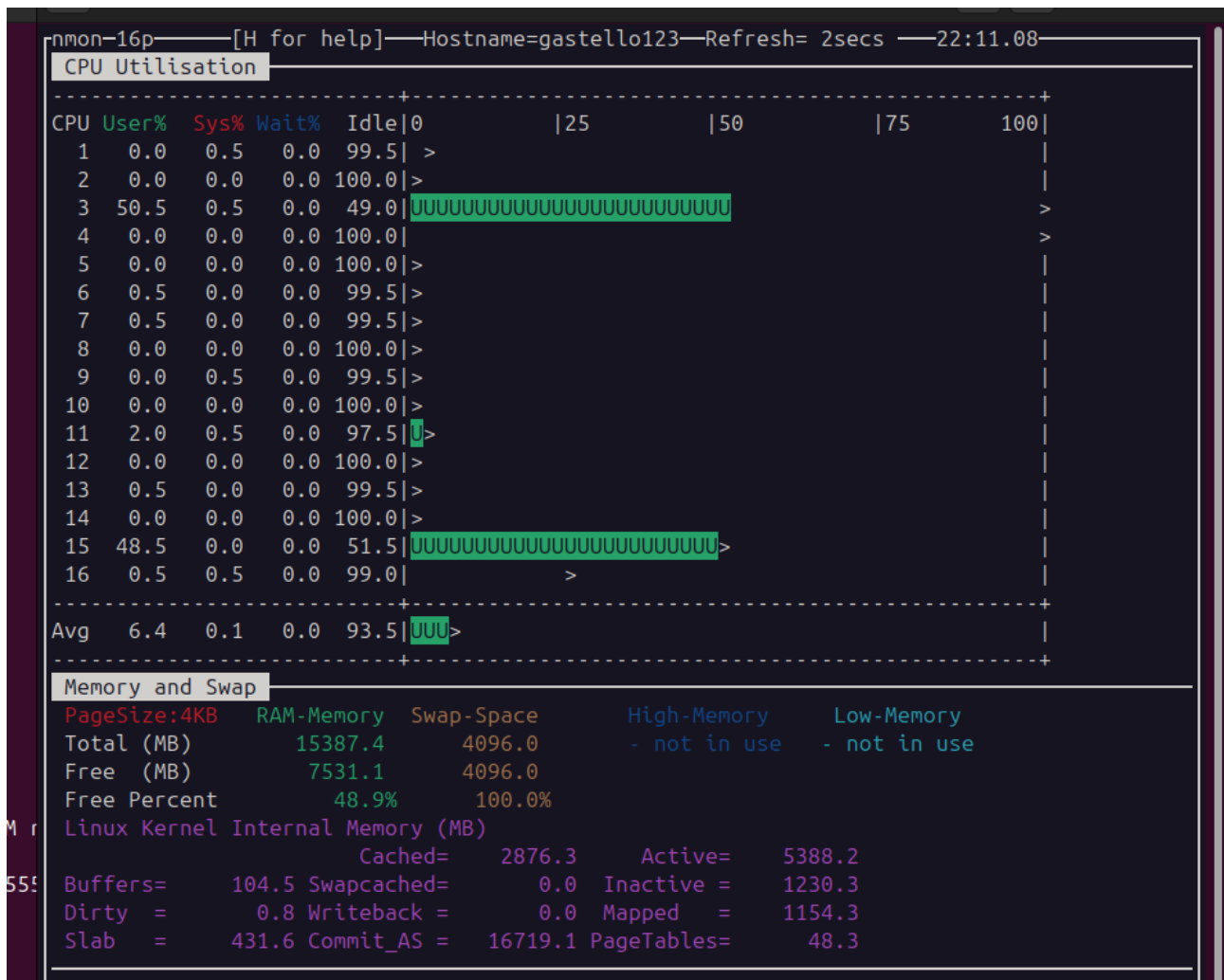
Use these keys to toggle statistics on/off:
c = CPU          l = CPU Long-term      - = Faster screen updates
C = " WideView  U = Utilisation          + = Slower screen updates
m = Memory       V = Virtual memory      j = File Systems
d = Disks        n = Network              . = only busy disks/procs
r = Resource     N = NFS                  h = more options
k = Kernel       t = Top-processes       q = Quit
```

С выключенным QEMU:



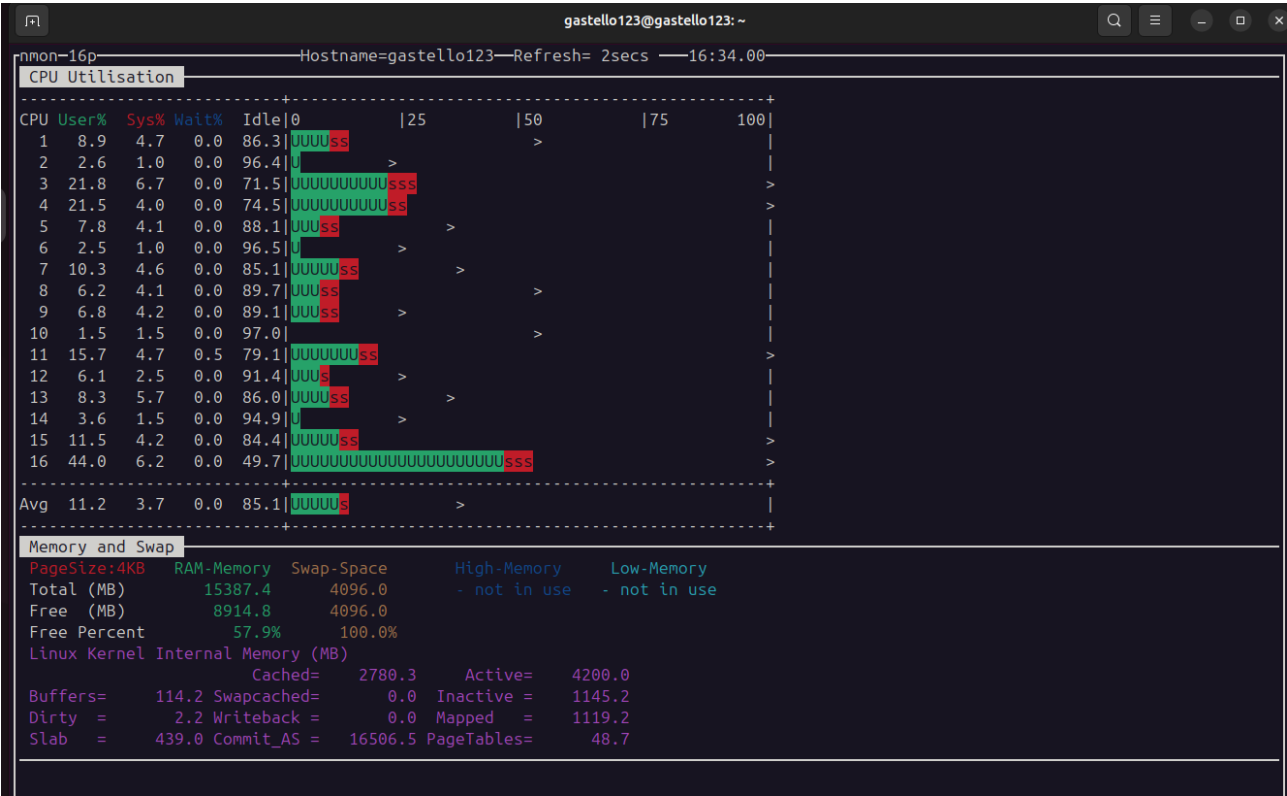
С включенным QEMU:





Нагружаем QEMU с помощью locust (тестирующий класс точно такой же как в лабораторной работе №6, только с другим хостом).

Первый тест — 500 пользователей, 1 запрос в секунду:



Нагрузка настолько сильная, что в какой-то момент QEMU просто закрывается:

```
gastello123@gastello123:~$ Perf stat -e cycles,instructions,cache-misses qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio
unterminated string
unterminated string

Performance counter stats for 'qemu-system-arm -M romulus-bmc -monitor telnet:127.0.0.1:5555,server,nowait -serial mon:stdio':

      389293421078      cycles
      931239395696      instructions
      212065522         cache-misses

      89.850152082 seconds time elapsed

      83.934452000 seconds user
      8.573117000 seconds sys
```

Также в логах locust в ходе теста появлялись новые интересные предупреждения:

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

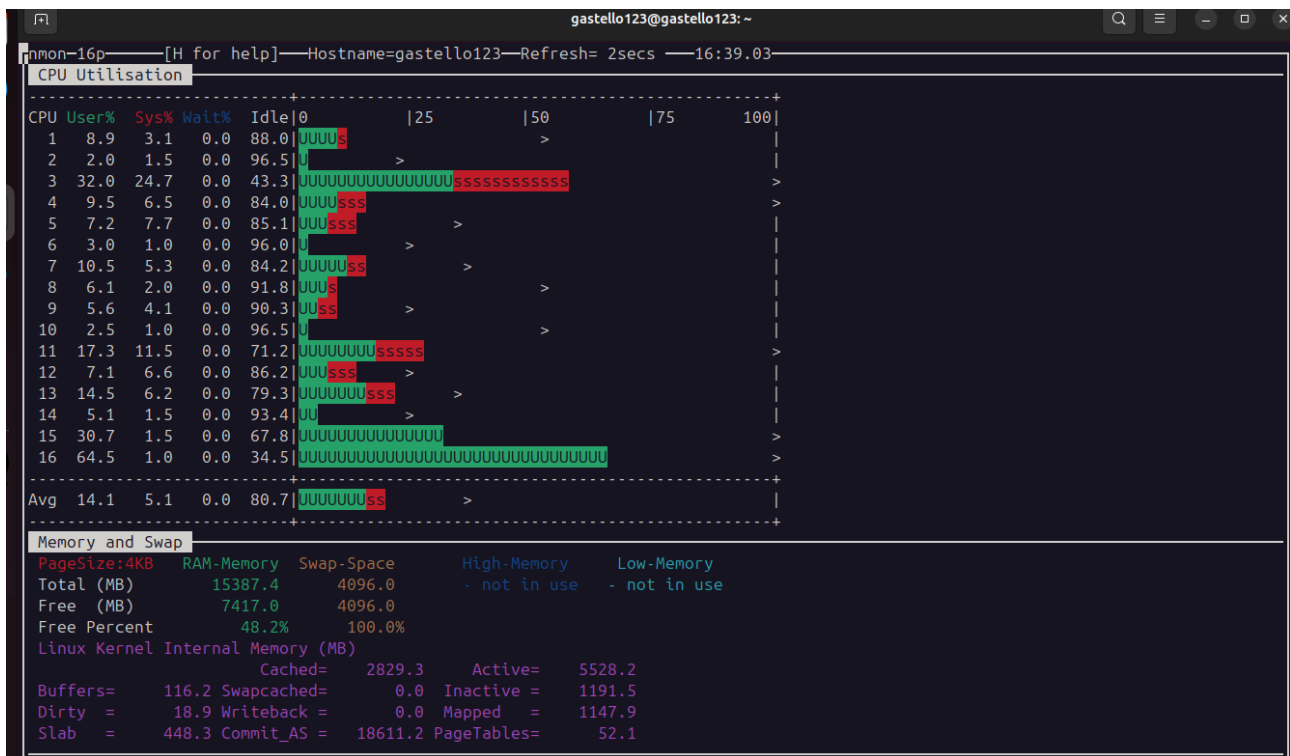
Logs

- [2025-10-30 16:33:22,641] gastello123/INFO/locust.main: Starting web interface at http://0.0.0.0:8089
- [2025-10-30 16:33:22,649] gastello123/INFO/locust.main: Starting Locust
- [2025-10-30 16:33:40,227] gastello123/INFO/locust.runners: Ramping to 500 users at a rate of 1.00 per second
- [2025-10-30 16:33:47,659] gastello123/WARNING/root: CPU usage above 90%! This may constrain your throughput and may even give inconsistent response time measurements! See <https://docs.locust.io/en/stable/running-distributed.html> for how to distribute the load over multiple CPU cores or machines
- [2025-10-30 16:34:40,189] gastello123/WARNING/locust.runners: CPU usage was too high at some point during the test! See <https://docs.locust.io/en/stable/running-distributed.html> for how to distribute the load over multiple CPU cores or machines

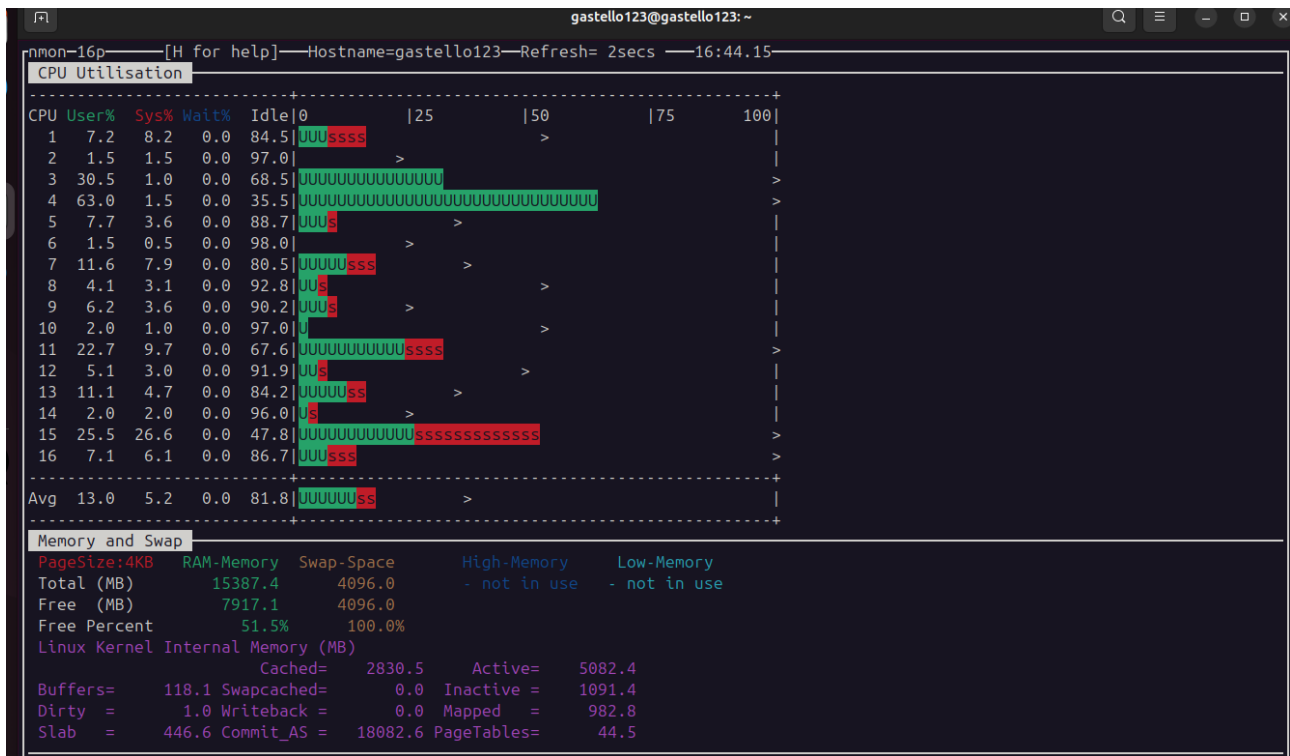
(html подробности, к сожалению, не прогрузились)

Второй тест — 100 пользователей, 1 запрос в секунду:

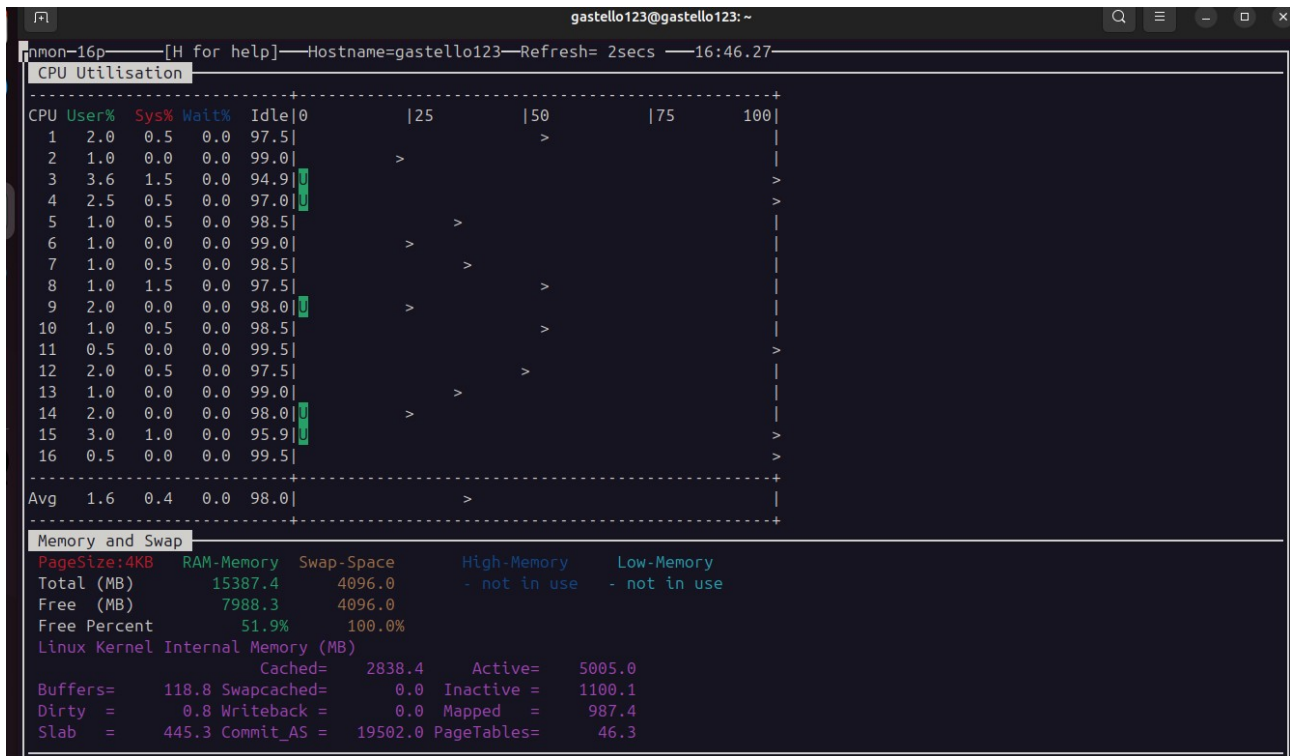




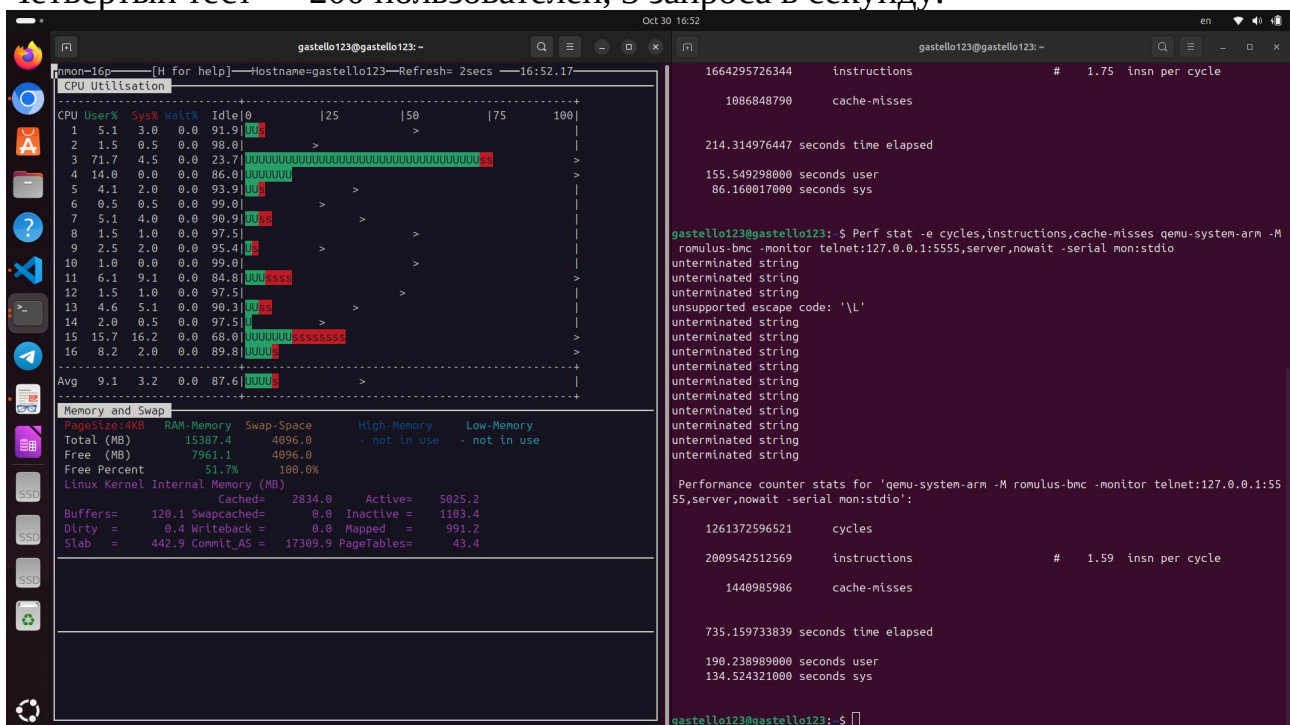
Третий тест — 200 пользователей, 2 запроса в секунду:



Отключаем нагрузку:



Четвёртый тест — 200 пользователей, 3 запроса в секунду:



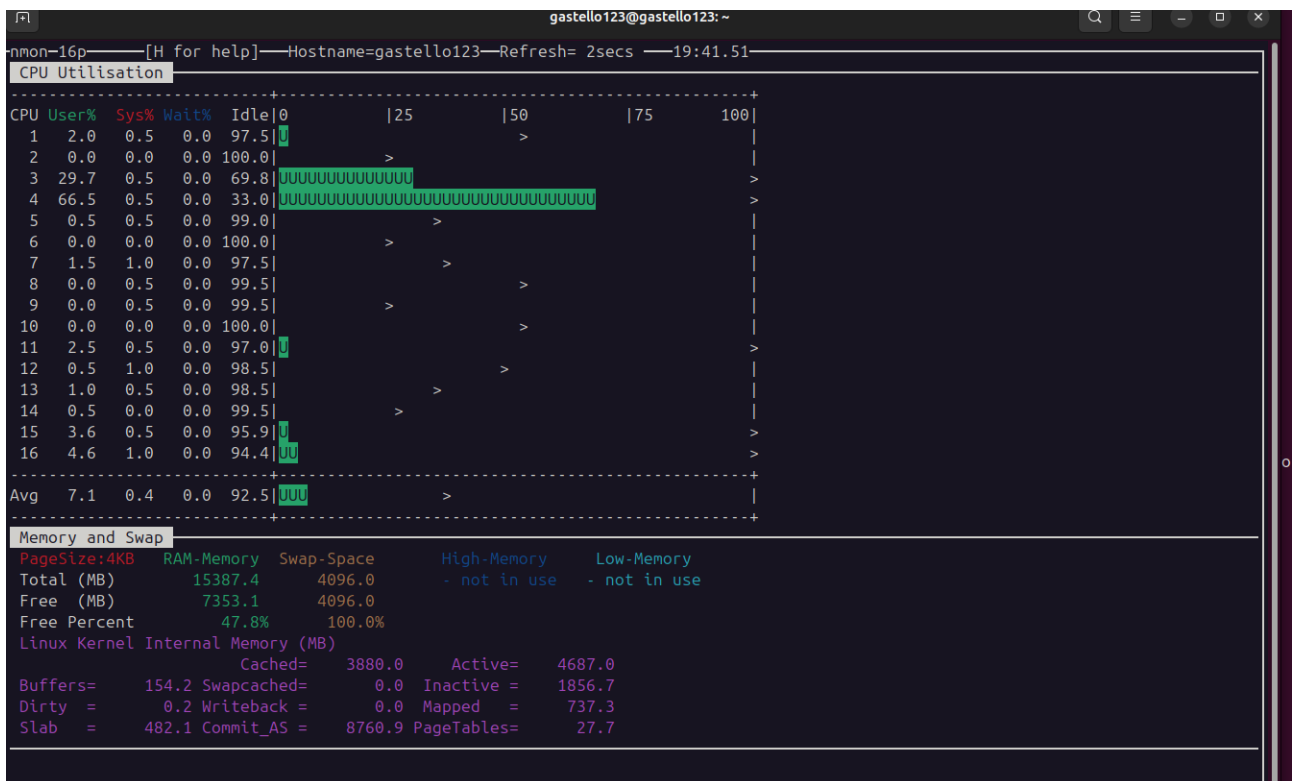
Снова достигли той нагрузки, при которой QEMU падает.

Попробуем оптимизировать потребление ресурсов QEMU за счёт параметров запуска:

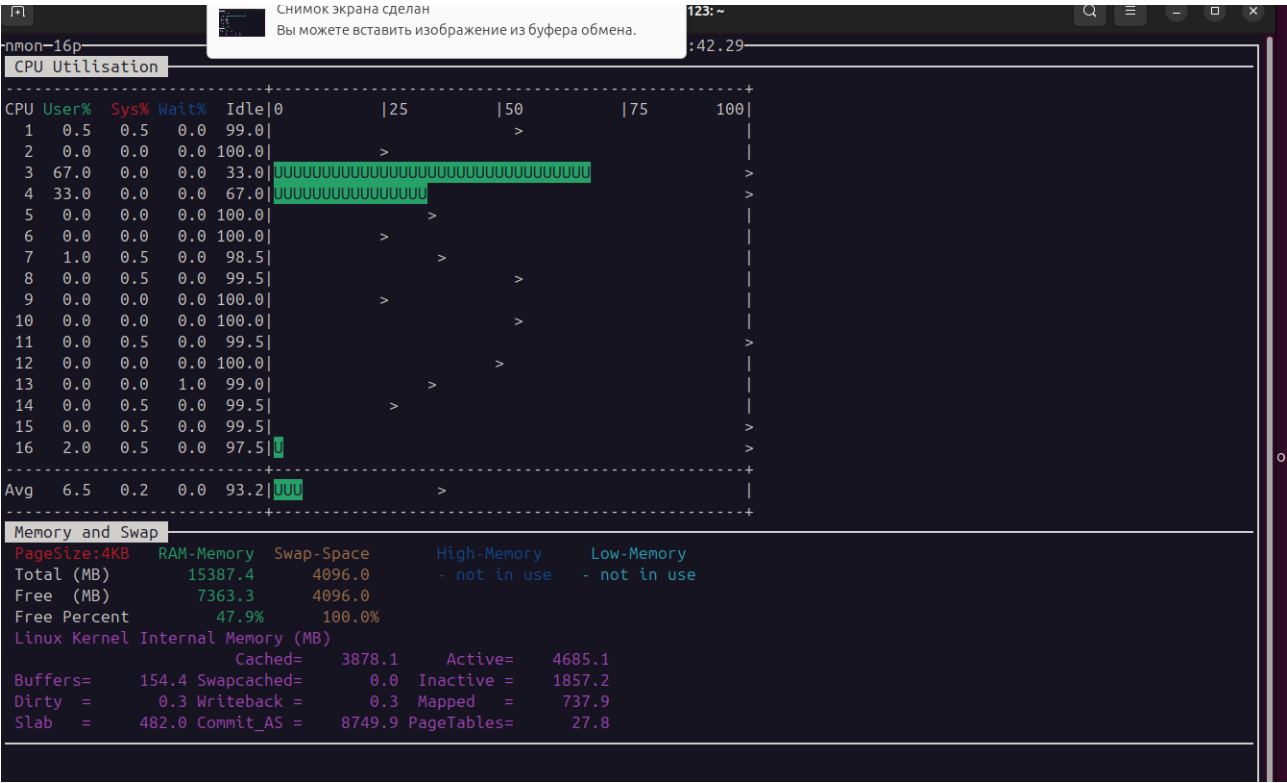
```
gastello123@gastello123:~$ Perf stat -e cycles,instructions,cache-misses qemu-system-arm -M romulus-bmc -cpu arm1026 -m 256 -nographic -monitor telnet:127.0.0.1:5555,server,nowait -ser ial mon:stdio
unterminated string
unterminated string
unterminated string
unterminated string
unterminated string
```

Самое очевидное: `-nographic` — отключение графического интерфейса, поскольку он не приносит нам никакой пользы.

До:



После:



Выигрыш минимальный.

Ещё одна простая идея: `-m 128M` — указываем конкретное количество памяти, которое нам нужно для работы (по умолчанию там 512 МБ, даже больше)

До:

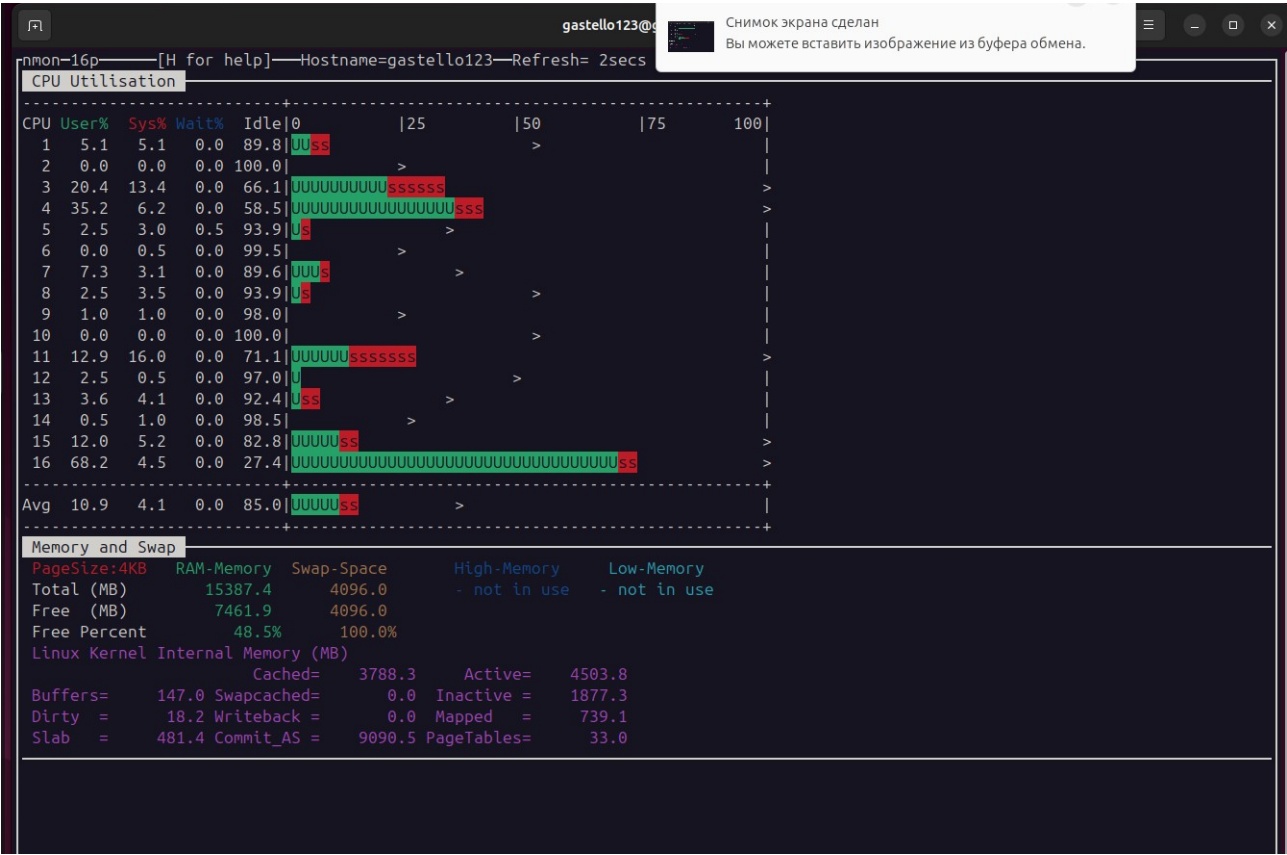
```
(qemu) info memdev
memory backend: ram
  size: 536870912
  merge: true
  dump: true
  prealloc: false
  share: false
  reserve: true
  policy: default
  host nodes:
```

После:

```
(qemu) info memdev
memory backend: ram
  size: 134217728
  merge: true
  dump: true
  prealloc: false
  share: false
  reserve: true
  policy: default
  host nodes:
```

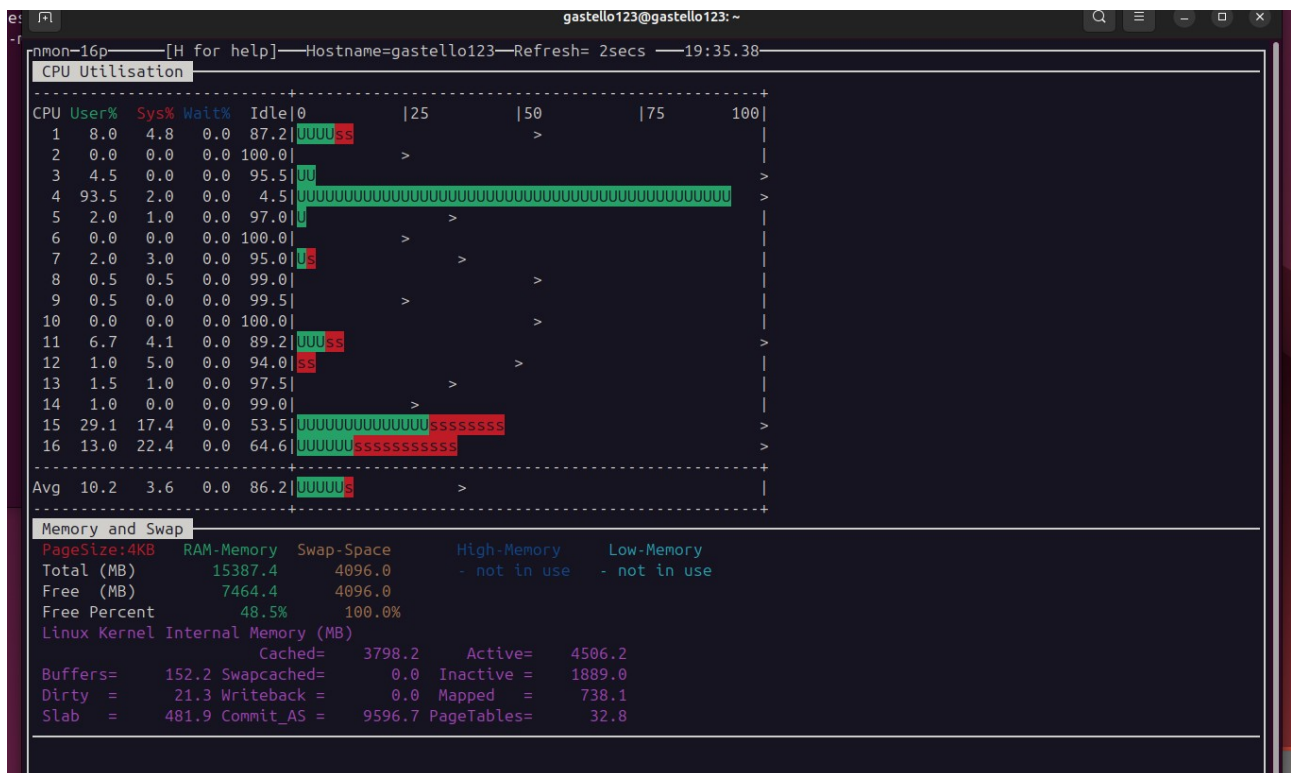
```
(qemu)
```

Данные берутся под тестом.



Как будто это возымело даже негативный эффект.

Ради интереса, попробуем поставить 1024 МБ:



Похоже этот флаг лучше не ставить.

Ещё я хотел добавить `-cpu host`, чтобы QEMU мог использовать возможности реального процессора хоста и не эмулировать инструкции, но у меня процессор AMD, который `qemu-system-arm` не поддерживает, поэтому это не рабочий вариант.