

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИТМО»**

Факультет безопасности информационных технологий

Дисциплина:
«Операционные системы»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3

Выполнили:

Нгуен Хонг Хань N3249

(подпись)

Проверил:

Савков Сергей Витальевич

(подпись)

Санкт-Петербург

2022г.

Задание

Простой вариант: Найти и скомпилировать программу `linpack` для оценки производительности компьютера (Flops) и протестировать ее при различных режимах работы ОС:

1. С различными приоритетами задачи в планировщике
2. С наличием и отсутствием привязки к процессору
3. Провести несколько тестов, сравнить результаты по 3 сигма или другим статистическим критериям

Усиленный вариант: То же самое, плюс изменить параметры на уровне ядра

Повлиять на настройки имеющегося планировщика

1. Теория

- FLOPS – это единица измерения вычислительной мощности компьютеров в операциях с плавающей точкой, которой часто пользуются, чтобы померить у кого больше.
- **Linpack**
Тесты LINPACK служат для измерения вычислительной производительности компьютеров при обработке чисел с плавающей запятой. Производительность теста LINPACK предоставить данные для уточнения пиковой производительности, предоставляемой производителем компьютера. Тесты измеряют скорость решения компьютером плотной системы линейных уравнений (СЛАУ) $Ax=b$, где A является матрицей размера n на n .
- **Приоритет задачи в планировщике**
В ядре Linux используются два разных диапазона приоритетов. Первый из них определяет значение параметра **nice** — число в диапазоне от -20 до $+19$, стандартное значение которого равно нулю. Большшему значению параметра **nice** соответствует меньший приоритет. Процессам с низким значением параметра **nice** (большим приоритетом) выделяется большая часть процессорного времени по сравнению с процессами, имеющими высокое значение параметра **nice**.
- **taskset** используется для того, чтобы пользователь мог сам устанавливать для планировщика CPU affinity, которое «связывает» процесс с заданным набором ЦП.
- Правило трёх сигм гласит: вероятность того, что любая случайная величина отклонится от своего среднего значения менее чем на 3 сигм.

2. Найти и скомпилировать программу `linpack` для оценки производительности компьютера (Flops).

```
hanhnguyen26@ubuntu:~/Downloads$ sudo unzip linpack-master.zip
Archive:  linpack-master.zip
0cd0fab81e16beee0b4fea677b52467b6de86d43
  creating: linpack-master/
  inflating: linpack-master/.gitignore
  inflating: linpack-master/CMakeLists.txt
  inflating: linpack-master/Dockerfile
  inflating: linpack-master/LICENSE
  inflating: linpack-master/Makefile
  inflating: linpack-master/README.md
  inflating: linpack-master/linpack.c
hanhnguyen26@ubuntu:~/Downloads$ ls
1.jpg  linpack-master  linpack-master.zip  OS-spr22-main
hanhnguyen26@ubuntu:~/Downloads$ cd linpack-master
```

```

hanhnguyen26@ubuntu:~/Downloads/linpack-master$ ls
CMakeLists.txt Dockerfile LICENSE linpack.c Makefile README.md
hanhnguyen26@ubuntu:~/Downloads/linpack-master$ sudo make
cc -Wall -Wextra -O3 -std=c11 -pedantic -march=native linpack.c -o linpack
strip linpack
hanhnguyen26@ubuntu:~/Documents/GitHub/OS-spring2022/Lab 3/linpack$ ./linpack
Memory required: 315K.

LINPACK benchmark, Double precision.
Machine precision: 15 digits.
Array size 200 X 200.
Average rolled and unrolled performance:

```

Reps	Time(s)	DGEFA	DGESL	OVERHEAD	KFLOPS
2048	0.57	71.78%	2.98%	25.24%	6579550.257
4096	1.17	71.85%	3.14%	25.01%	6418038.563
8192	2.28	71.72%	2.94%	25.34%	6620908.282
16384	4.55	71.69%	2.94%	25.38%	6627074.309
32768	9.10	71.70%	2.94%	25.36%	6625950.230
65536	18.20	71.69%	2.94%	25.37%	6627017.705

3. Протестировать с различными приоритетами задачи в планировщике. Код сбора информации

```

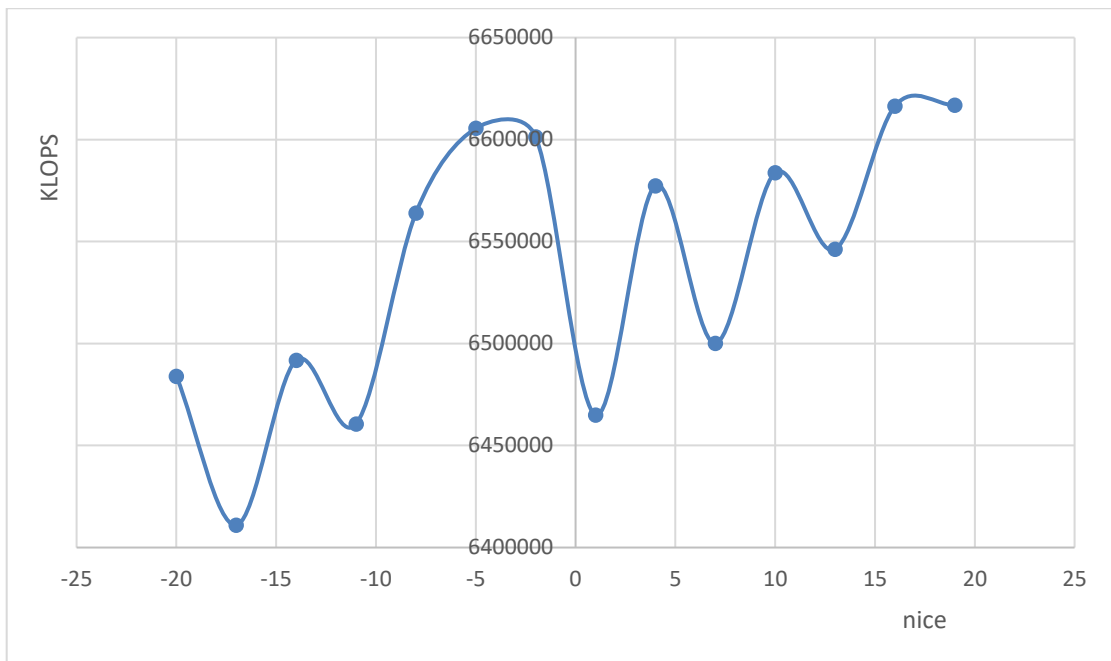
for ((i = -20; i <= 19; i = $i + 3))
do
echo $i >> out.txt
sudo nice -n $i ./linpack >> out.txt
done

```

Результат

nice	Матожидание	Сигма	Левая граница	Правая граница
-20	6484022	145060	6048842	6919203
-17	6410992	236857	5700420	7121564
-14	6491828	81690	6246758	6736897
-11	6460541	114937	6115730	6805351
-8	6563980	2235	6557275	6570685
-5	6605553	3364	6595460	6615645
-2	6601382	3827	6589902	6612863
1	6464996	58911	6288263	6641729
4	6577408	78970	6340497	6814319
7	6500149	64450	6306798	6693501
10	6583732	2605	6575917	6591547
13	6546214	4128	6533828	6558599
16	6616491	1181	6612949	6620033
19	6616849	2620	6608989	6624709

Правило трёх сигм выполняется



4. Протестировать с наличием и отсутствием привязки к процессору.

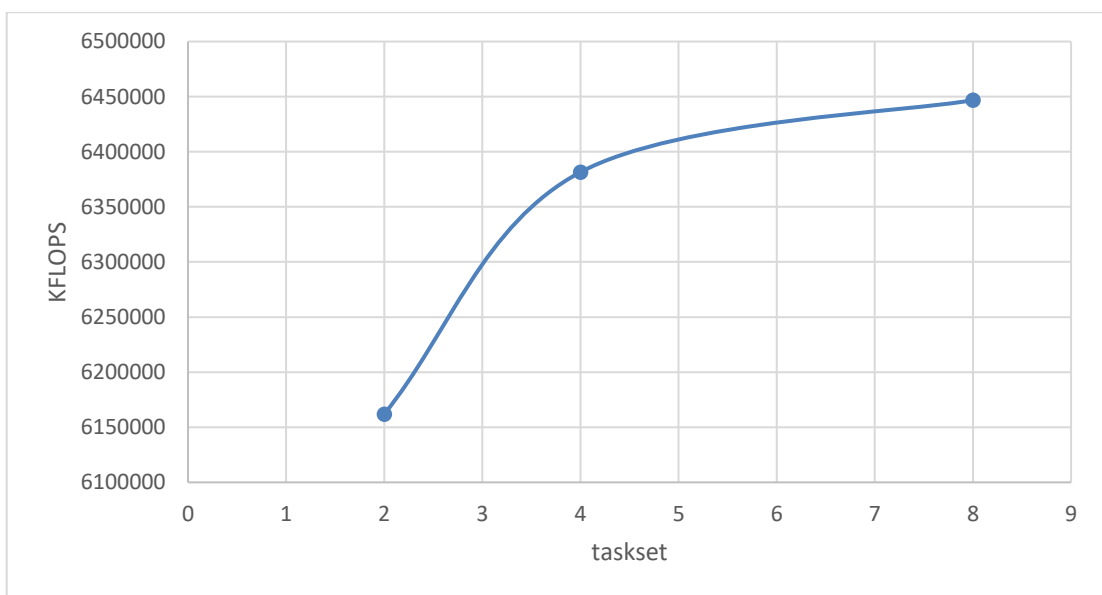
Код сбора информации

```
for ((i = 2; i <= 8 ; i = $i *2))
do
echo $i >> out.txt
sudo taskset -c 1-$i ./linpack >> out.txt
done
```

Результат

Taskset	Матожидание	Сигма	Левая граница	Правая граница
2	6161986	291490	5287517	7036456
4	6381506	70796	6169116	6593895
8	6446860	111261	6113076	6780644

Правило трёх сигм выполняется



5. Повлиять на настройки имеющегося планировщика.

Просмотреть все параметры планировщика

```
hanhnguyen26@yuety25:~/Documents/GitHub/OS-spring2022/Lab 3/linpack$ sudo sysctl -A | grep "sched" | grep -v "domain"
[sudo] password for hanhnguyen26:
kernel.sched_autogroup_enabled = 1
kernel.sched_cfs_bandwidth_slice_us = 5000
kernel.sched_child_runs_first = 0
kernel.sched_deadline_period_max_us = 4194304
kernel.sched_deadline_period_min_us = 100
kernel.sched_energy_aware = 1
kernel.sched_rr_timeslice_ms = 100
kernel.sched_rt_period_us = 1000000
kernel.sched_rt_runtime_us = 950000
kernel.sched_schedstats = 0
kernel.sched_util_clamp_max = 1024
kernel.sched_util_clamp_min = 1024
kernel.sched_util_clamp_min_rt_default = 1024
```

Квант времени (timeslice) - это числовое значение, определяющее, как долго может выполняться процесс до того момента, пока он не будет вытеснен.

energy_aware - способность прогнозировать влияние своих решений на энергию, потребляемую центральными процессорами.

Изменим значение переменной kernel.sched_rr_timeslice_ms с 10 по 100 с шагом 10.

При этом kernel.sched_energy_aware = 0 или 1.

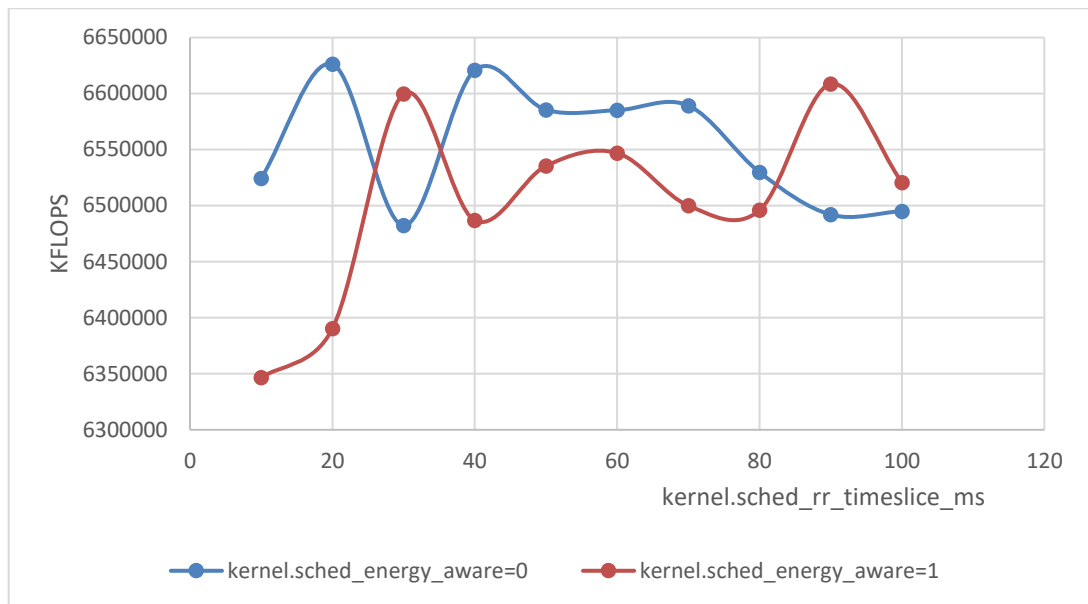
Код сбора информации

```
for ((i = 10; i <= 100 ; i = $i + 10))
do
echo $i >> out1.txt
sudo sysctl kernel.sched_rr_timeslice_ms=$i
sudo ./linpack >> out1.txt
done
```

Результат

	timeslice	Матожидание	Сигма	Левая граница	Правая граница
kernel.sched_energy_aware=0	10	6524396	3609	6513571	6535222
	20	6626223	7038	6605109	6647336
	30	6482402	92888	6203737	6761066
	40	6620939	3143	6611511	6630367
	50	6585549	47125	6444173	6726924
	60	6585194	2143	6578767	6591622
	70	6589304	84306	6336385	6842223
	80	6529786	106029	6211699	6847873
	90	6492009	175862	5964423	7019596
	100	6495061	67674	6292040	6698083
kernel.sched_energy_aware=1	10	6346578	240273	5625760	7067395
	20	6390307	2166	6383810	6396804
	30	6599742	4542	6586115	6613368
	40	6486850	1858	6481276	6492424
	50	6535471	3852	6523915	6547028
	60	6547023	4021	6534959	6559086
	70	6500064	1886	6494405	6505722
	80	6495950	1574	6491227	6500673
	90	6608672	1902	6602965	6614378
	100	6520721	1356	6516655	6524788

Правило трёх сигм выполняется



Вывод: В ходе лабораторной работы я познакомилась с Linpack. Работа программы была протестирована с различными приоритетами задачи в планировщике и с наличием или отсутствием привязки к процессу. Также была выполнена усложненная часть задания, в которой было необходимо повлиять на настройки имеющегося планировщика. Планировщик определяет, когда и как долго выполняются процессы. Таким образом, поведение планировщика сильно влияет на производительность системы.