

Задание: 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
4. (Усиленный вариант) Повлиять на настройки имеющегося планировщика.

Task: Find and compile the linpack program to evaluate computer performance (Flops) and test it under various OS operating modes:

1. With different task priorities in the scheduler
2. With and without processor affinity
3. Conduct multiple tests, compare results by 3 sigma or other statistical criteria
4. (Enhanced version) Influence the settings of the existing scheduler.

Ход работы

1. Тест программы без изменений:

```
travis@ubuntu:~/Рабочий стол/linpack-master$ ./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	0.62	77.54%	2.99%	19.47%	5617454.950
	4096	1.21	77.47%	2.94%	19.60%	5786405.743
	8192	2.51	77.51%	2.96%	19.52%	5571911.432
	16384	4.82	77.51%	2.94%	19.55%	5806927.253
	32768	9.75	77.54%	2.94%	19.52%	5734012.780
	65536	20.18	77.59%	2.97%	19.44%	5535362.764

2. Тест программы с различными приоритетами в планировщике:

Изменение приоритета выполнялось с помощью утилиты nice. Значение опции -n -20 выставляет наивысший приоритет выполняемой задаче, а 19 – самый низкий.

```
travis@ubuntu:~/Рабочий стол/linpack-master$ sudo nice -n -20 ./linpack
[sudo] пароль для travis:
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	0.59	77.34%	2.97%	19.69%	5905061.435
4096	1.26	77.56%	2.96%	19.48%	5551345.983
8192	2.44	77.54%	2.93%	19.53%	5734530.836
16384	4.91	77.58%	2.95%	19.47%	5690553.525
32768	9.45	77.57%	2.94%	19.49%	5915836.887
65536	21.45	77.75%	3.02%	19.24%	5194605.472

```
travis@ubuntu:~/Рабочий стол/linpack-master$ sudo nice -n 19 ./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	0.61	77.84%	2.94%	19.22%	5706317.163
4096	1.21	77.58%	2.94%	19.49%	5778411.015
8192	2.38	77.56%	2.94%	19.50%	5881743.821
16384	4.72	77.50%	2.94%	19.56%	5921609.084
32768	9.47	77.50%	2.94%	19.55%	5904131.750
65536	19.06	77.51%	2.94%	19.55%	5868671.865

3. Тест программы с наличием и отсутствием привязку к процессору:

Привязка к ядру выполняется с помощью утилиты taskset. Тест проводился с наивысшим приоритетом в планировщике.

```
travis@ubuntu:~/Рабочий стол/linpack-master$ sudo taskset 0x1 nice -n -20 ./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	0.59	77.49%	2.98%	19.53%	5951959.730
4096	1.19	77.55%	2.92%	19.53%	5856840.666
8192	2.37	77.48%	2.92%	19.60%	5894761.404
16384	4.76	77.59%	2.93%	19.48%	5875373.162
32768	9.68	77.65%	2.94%	19.41%	5770041.772
65536	19.30	77.55%	2.94%	19.51%	5794952.400

Также программа была протестирована при привязке к ядру, которое было предварительно очищено от сторонних процессов с помощью команды:

```
for c in `ps aux | grep -v "linpack" | awk '{print $2}'`; do taskset -a -p 0x2 $c; done.
```

4. С помощью утилиты `sysctl` можно просматривать и изменять настройки планировщика:

```
root@ubuntu:/home/travis/Рабочий стол/linpack-master# sysctl -a | grep sched
kernel.sched_autogroup_enabled = 1
kernel.sched_cfs_bandwidth_slice_us = 5000
kernel.sched_child_runs_first = 0
kernel.sched_domain.cpu0.domain0.busy_factor = 32
kernel.sched_domain.cpu0.domain0.cache_nice_tries = 1
kernel.sched_domain.cpu0.domain0.flags = 2071
kernel.sched_domain.cpu0.domain0.imbalance_pct = 125
kernel.sched_domain.cpu0.domain0.max_interval = 4
kernel.sched_domain.cpu0.domain0.max_newidle_lb_cost = 13771
kernel.sched_domain.cpu0.domain0.min_interval = 2
```

Изменяем значения следующих параметров:

`kernel.sched_latency_ns = 12000000`. Увеличение этой переменной увеличивает временной интервал задачи, привязанной к ЦП. Повышаем значение до 20000000 нс.

`kernel.sched_autogroup_enabled = 1`. Этот параметр отвечает за группирование задач. Меняем его значение на 0.

`kernel.sched_min_granularity_ns = 1500000`. Это начальное значение периода планировщика. Период планировщика — это период времени, в течение которого все выполняемые задачи должны запускаться хотя бы один раз. Заносим в переменную 1000000 нс.

```
root@ubuntu:/home/travis/Рабочий стол/linpack-master# sysctl -w kernel.sched_autogroup_enabled=0
kernel.sched_autogroup_enabled = 0
root@ubuntu:/home/travis/Рабочий стол/linpack-master# sysctl -w kernel.sched_min_granularity_ns=1000000
kernel.sched_min_granularity_ns = 1000000
root@ubuntu:/home/travis/Рабочий стол/linpack-master# sysctl -w kernel.sched_latency_ns=20000000
kernel.sched_latency_ns = 20000000
root@ubuntu:/home/travis/Рабочий стол/linpack-master#
```

Таблица значений показателя FLOPS для тестов

Желтым выделены математические ожидания для каждого теста, зеленым — среднеквадратичные отклонения.

Обычный тест	С наивысшим приоритетом	С наименьшим приоритетом	С привязкой к ядру и наивысшим приоритетом	С привязкой, наивысшим приоритетом и очисткой ядра	С привязкой, наивысшим приоритетом, очисткой ядра и настройками планировщика
5617454,950	5905061,435	5706317,163	5951959,730	5954139,543	5963254,059
5786405,743	5551345,983	5778411,015	5856840,666	5926306,759	5872548,569
5571911,432	5734530,836	5881743,821	5894761,404	5638398,196	5803257,909
5806927,253	5690553,525	5921609,084	5875373,162	5784287,514	5853913,541
5734012,780	5915836,887	5904131,750	5770041,772	5881329,493	5906293,719
5535362,764	5194605,472	5868671,865	5794952,400	5086422,053	5790240,049
3342285,431	5981223,613	5870232,040	5711253,344	5759095,336	5939654,268
4201452,376	5971071,536	5887207,031	5666168,059	5698051,411	5873572,592
4388248,853	5933304,679	5904692,625	5743879,213	5695870,478	5863831,266

4674021,619	5887040,677	5404345,040	5750154,376	5742738,680	5872061,211
5106732,721	5872714,106	5357735,918	5798451,354	5673909,378	5566260,037
5066549,517	5896928,105	5681740,744	5815275,038	5666666,079	5630781,592
5781386,394	5409850,465	5713353,227	5405868,077	5915742,788	5776440,994
5708425,004	5444766,442	5669411,753	5534735,751	5813411,601	5721500,516
5697785,917	5846310,009	5643451,060	5484209,943	5751716,856	5770074,325
5690786,681	5851123,234	5419420,952	5262010,422	5771785,339	5856711,085
5814695,332	5889097,662	5590145,482	5314551,735	5685862,093	5798260,095
5789865,701	5841542,232	5686462,021	5082008,288	5713758,372	5923404,924
5963734,543	5927487,027	5776070,281	5353112,116	5790825,693	5856328,475
5944330,376	5916327,650	5720745,538	5496382,108	5609729,747	5911909,828
5912360,324	5934265,485	5688939,950	5306933,763	5767045,537	5836200,116
5860314,039	5919825,235	5772513,123	5518699,004	5747585,407	5768018,307
5930660,155	5850701,798	5851558,426	4123150,748	5823484,076	5823449,034
5886651,398	5785022,545	5827347,321	4918653,085	5743533,560	5890912,391
5943658,321	5781065,547	5642110,950	4786758,263	5221431,374	5857029,711
5899642,397	5739656,522	5793897,420	4990296,575	5642965,604	5849757,212
5909528,002	5871671,995	5758009,902	5449418,682	5664576,125	5906453,411
5912835,749	5880867,575	5710690,930	4354038,049	5804820,932	5888569,796
5916236,646	5886969,825	5864026,876	4333954,065	5787165,500	5495549,800
5786431,412	5833048,225	5883032,532	4905420,592	5742399,026	5458450,424
5539356,461	5804793,878	5739267,528	5374977,059	5716835,152	5810822,975
621066,048	181243,437	149150,624	493703,431	175826,297	124196,929

Вывод:

Наилучшие показатели FLOPS были получены в эксперименте с привязкой задачи к ядру, его предварительной очисткой от других процессов, наивысшим приоритетом и изменением настроек планировщика задач в системе. Математическое ожидание в этом случае получилось наибольшим, а среднеквадратичное отклонение – наименьшим. Неожиданно, тесты с привязкой к ядру и наивысшим приоритетом показали наихудший результат, математическое ожидание оказалось наименьшим. Наибольшее отклонение показателя FLOPS от среднего наблюдается в тестах с обычным запуском linpack без каких-либо настроек.

Таким образом, поставленные задачи и цель были выполнены.