

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

Санкт-Петербургский национальный исследовательский университет ИТМО

Мегафакультет трансляционных информационных технологий

Факультет информационных технологий и программирования

Лабораторная работа №5
«Управление памятью в ОС Linux»

По дисциплине «Операционные системы»

Выполнил студент группы М3215

Рагулин Антон Витальевич

Проверила

Тетерина Мария Олеговна



УНИВЕРСИТЕТ ИТМО

Санкт-Петербург
2025

Общий объем оперативной памяти – 1870900 kB

Объем раздела подкачки – 839676 kB

Размер страницы виртуальной памяти – 4096 kB

Объем свободной физической памяти в ненагруженной системе – 1461116 kB

Объем свободного пространства в разделе подкачки в ненагруженной системе – 839676 kB

Эксперимент №1

Первый этап:

```
[ 2248.103062] Out of memory: Killed process 1544 (bash) total-vm:2653136kB, anon-rss:1639752kB, file-rss:0kB, shmem-rss:0kB, UID:1000
[ 2248.190429] oom_reaper: reaped process 1544 (bash), now anon-rss:0kB, file-rss:0kB, shmem-rss:0kB
Killed
```

Последняя запись в report.log – 30000000

```
[ 1821.536089] Out of memory: Killed process 1850 (bash) total-vm:2634392kB, anon-rss:1667864kB, file-rss:0kB, shmem-rss:0kB, UID:1000
[ 1821.619799] oom_reaper: reaped process 1850 (bash), now anon-rss:0kB, file-rss:0kB, shmem-rss:0kB
```

Последняя запись в report.log – 30000000

Второй этап:

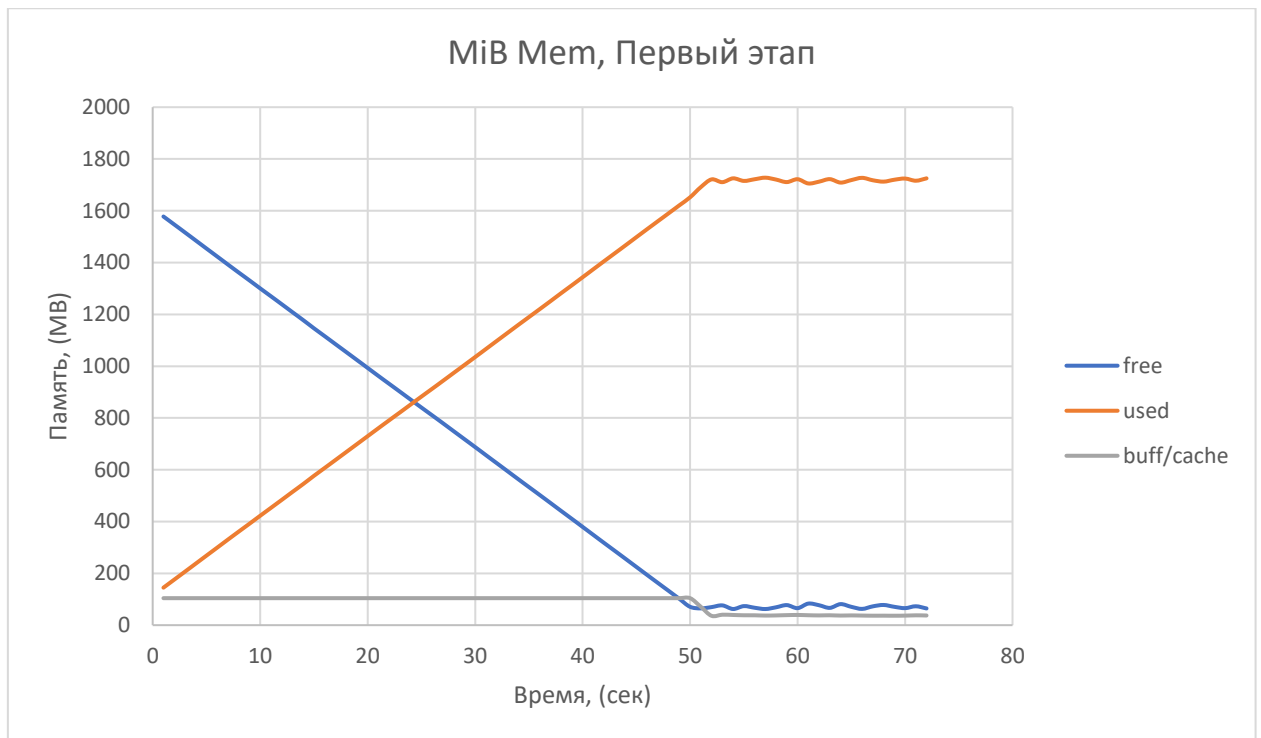
```
[ 7544.558426] [ 1669] 1000 1669 356747 199981 2482176 101206 0 bash
[ 7544.559022] Out of memory: Killed process 1669 (bash) total-vm:1426988kB, anon-rss:799924kB, file-rss:0kB, shmem-rss:0kB, UID:1000
[ 7591.779191] Out of memory: Killed process 1670 (bash) total-vm:2615384kB, anon-rss:1652520kB, file-rss:136kB, shmem-rss:0kB, UID:1000
[ 7591.894256] oom_reaper: reaped process 1670 (bash), now anon-rss:0kB, file-rss:0kB, shmem-rss:0kB
```

Последняя запись в report.log – 15000000

Последняя запись в report2.log – 30000000

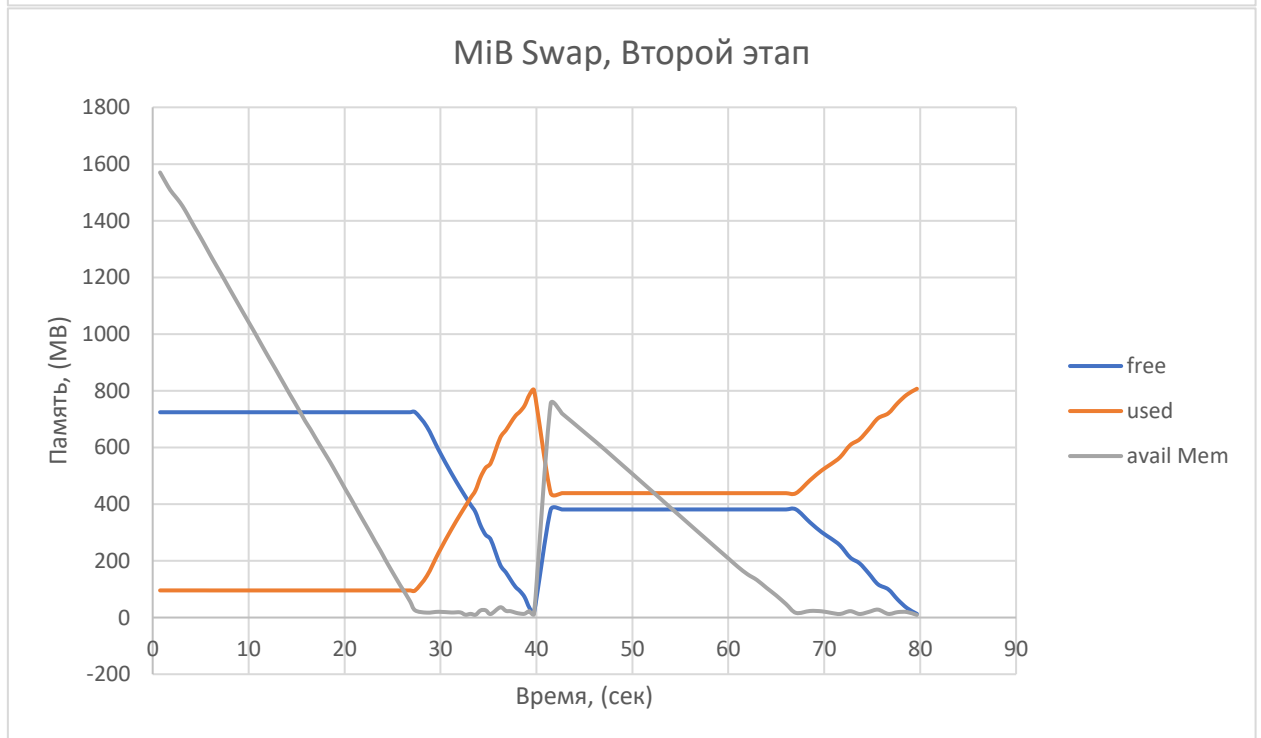
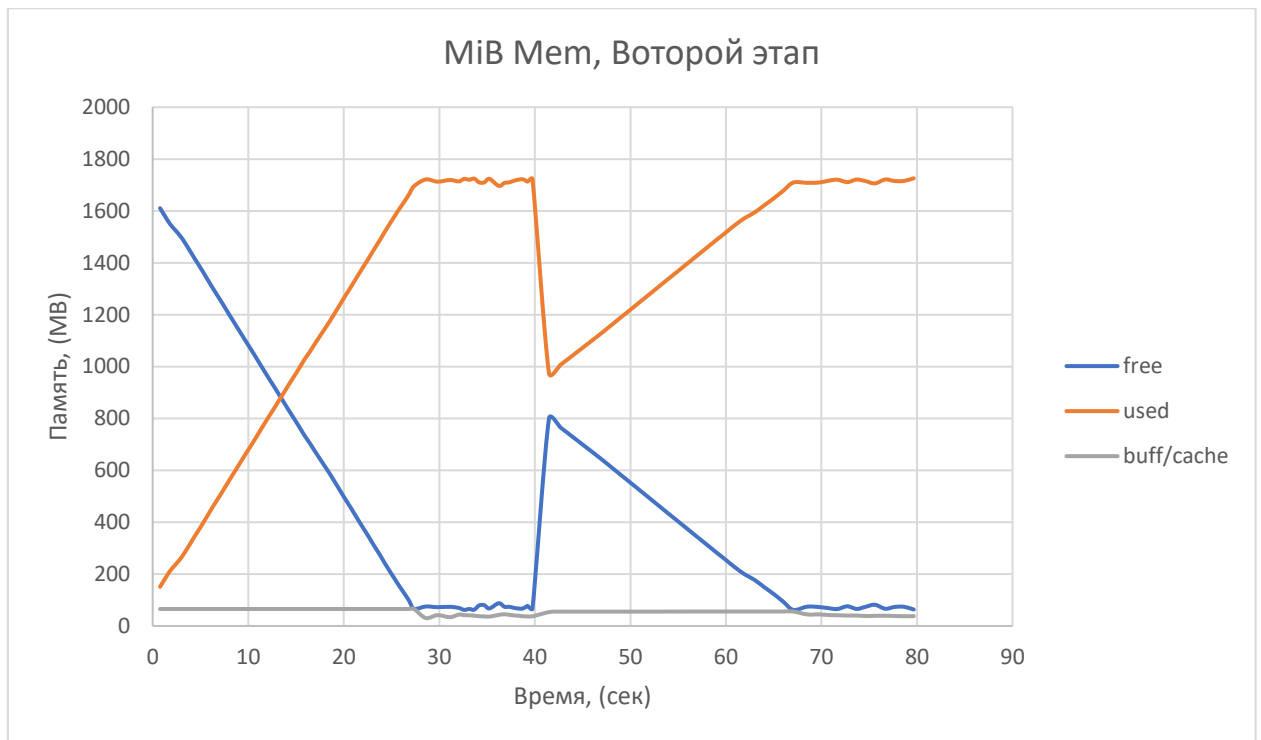
Результаты:

Графики первого этапа:



На первом графике можно заметить, что free и used противоположны, free растёт линейно до 52 секунд, на этом моменте он практически приблизился к общему объёму оперативной памяти системы, тогда на этом моменте исходя из второго графика ОС начинает использовать раздел подкачки и записывает данные туда до момента, когда его used практически равно объёму раздела подкачки, после чего процесс требует ещё выделение памяти, что невозможно из-за чего происходит аварийное завершение этого процесса.

Графики второго этапа:



Можно заметить из первого графика, что к моменту примерно 29 секунд запущенные процессы подходят к пределу оперативной памяти из-за чего ОС начинает использовать раздел подкачки и примерно к моменту 40 секунд практически полностью занимают память раздела подкачки, из-за чего при обращении одного из процессов к ОС с запросом памяти, происходит его аварийное завершение, это происходит в момент 40-41 секунд, когда видно, что освободилась память занимаемая одним из процессов, после чего использование раздела подкачки завершается и далее до 67 секунд используется только оперативная память. Далее на 67 секунде можно заметить, что used практически приблизился снова к общему размеру оперативной памяти и данные начинают записываться в раздел подкачки и когда приходит к почти полному использованию раздела подкачки процесс аварийно завершается. Так же из значений report.log и report2.log и графиков можно заметить, что аварийное завершение первого процесса произошло, когда сумма количества элементов в первом и втором массивах была примерно равна предельному значению из первого

этапа, а затем второй процесс записал в массив столько же элементов, сколько было максимально достигнуто на первом этапе .

Эксперимент №2

При $k=30$, максимальное n при котором все процессы завершаются без ошибок и аварийных остановок: $n=1600000$.

Вывод:

Можно заметить, что в первом случае $k=10$ $n=3000000$, получается, что произведение k на n равно крайнему количеству элементов массива из первого эксперимента, после которого происходила аварийная остановка процесса, поэтому при данных значениях все процессы завершаются корректно. Во втором же случае при $k=30$ максимальное значение n при котором не происходит аварийных остановок $n=1600000$, при этом их произведение намного больше размера массива из первого пункта первого эксперимента, но аварийной остановки не происходит потому, что при запуске следующих процессов успевают завершиться какое-то количество предыдущих из-за чего память освобождается для использования новыми процессами.