МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

Национальный исследовательский университет информационных технологий,   
механики и оптики

Мегафакультет трансляционных информационных технологий

Факультет информационных технологий и программирования

**Лабораторная работ №5**

По дисциплине «Операционные системы»

Управление памятью в OC Linux

Выполнил студент группы М3212:  
Шарифов Сардорбек Дилшод угли

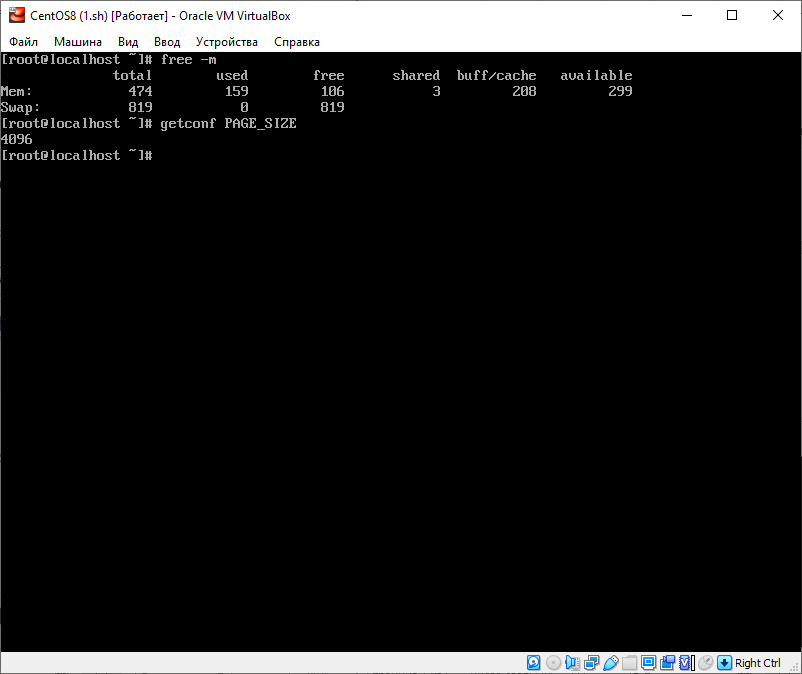
Преподователь:  
Дюкарева Вероника Макссимовна

**Санкт-Петербург  
2020**

**Отчет**

Данные о текущей конфигурации операционной системы:

* Общий объем оперативной памяти – 512Мб
* Объем раздела подкачки – 819Мб
* Размер страницы виртуальной памяти – 4Кб
* Объем свободной физической памяти в ненагруженной системе – 130Мб
* Объем свободного пространства в разделе подкачки в ненагруженной системе – 819Мб

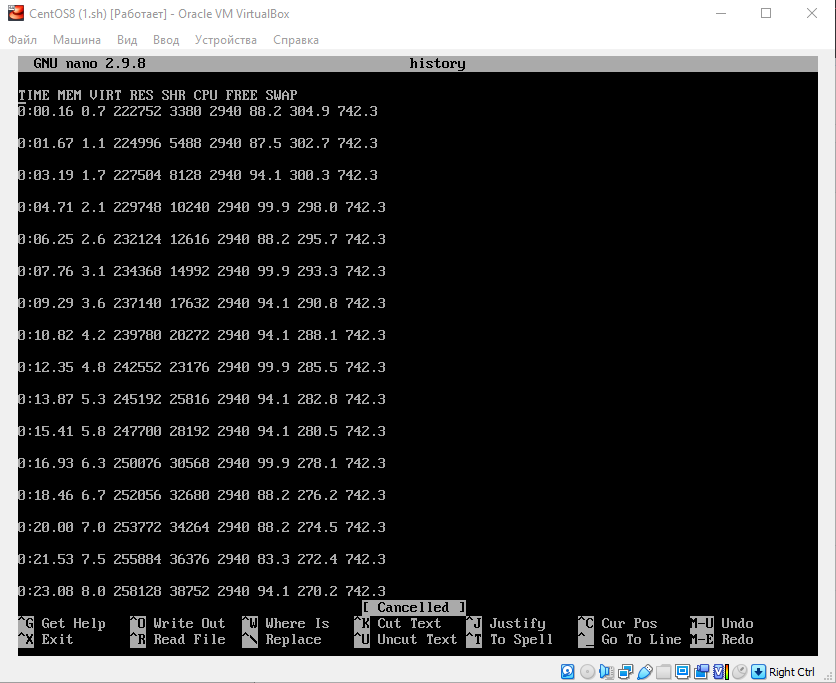
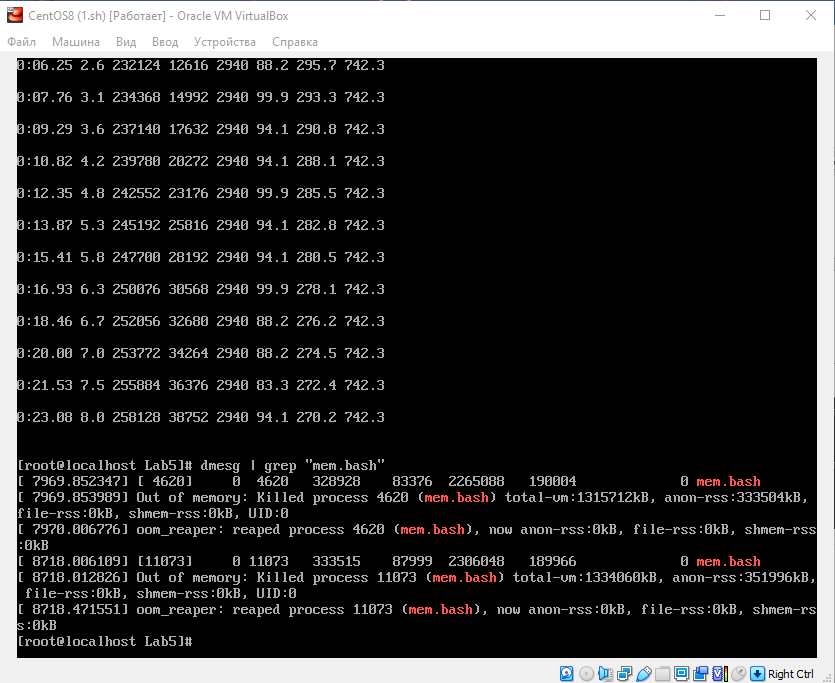
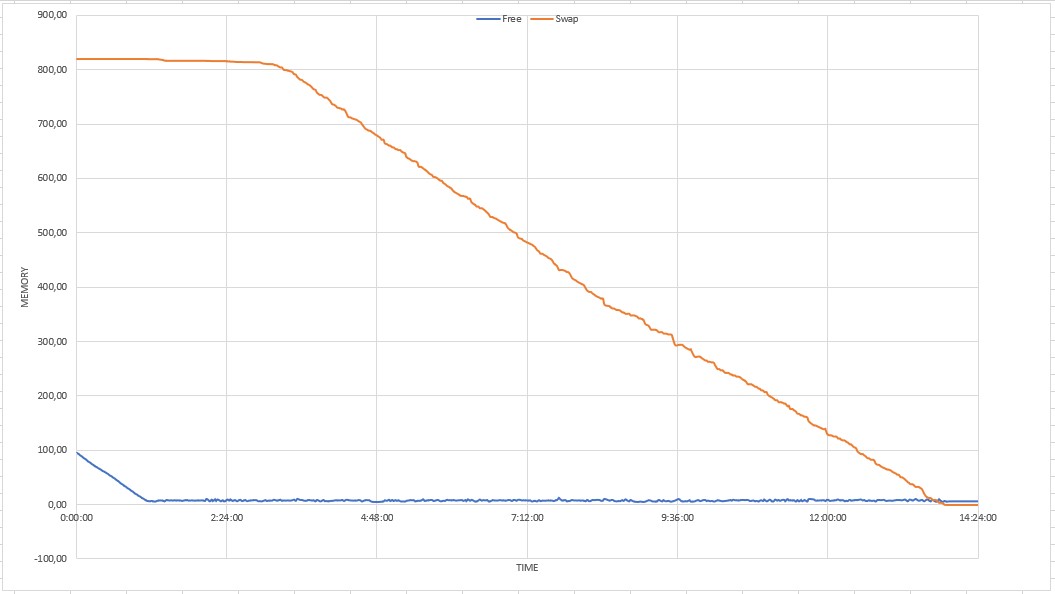


**Эксперимент №1**

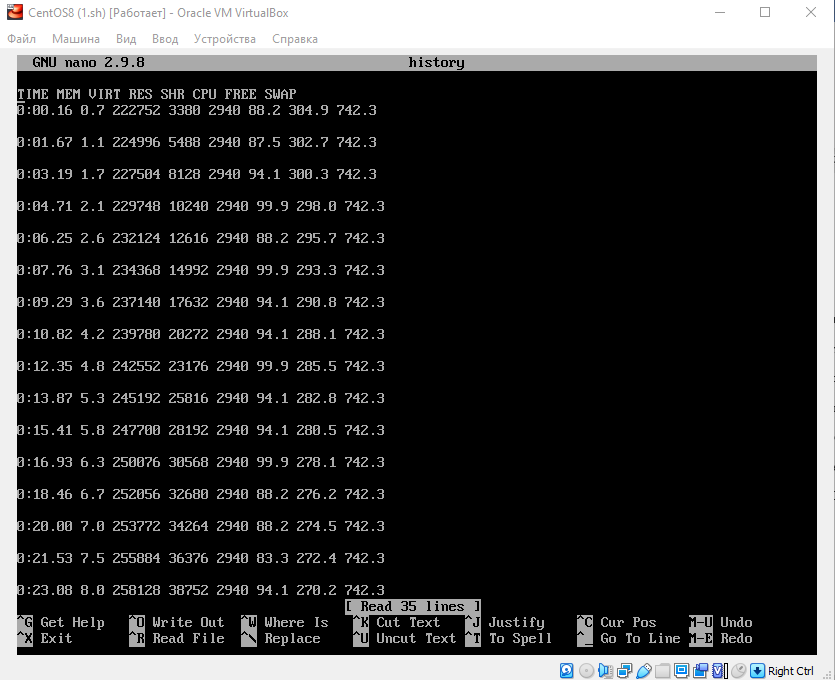
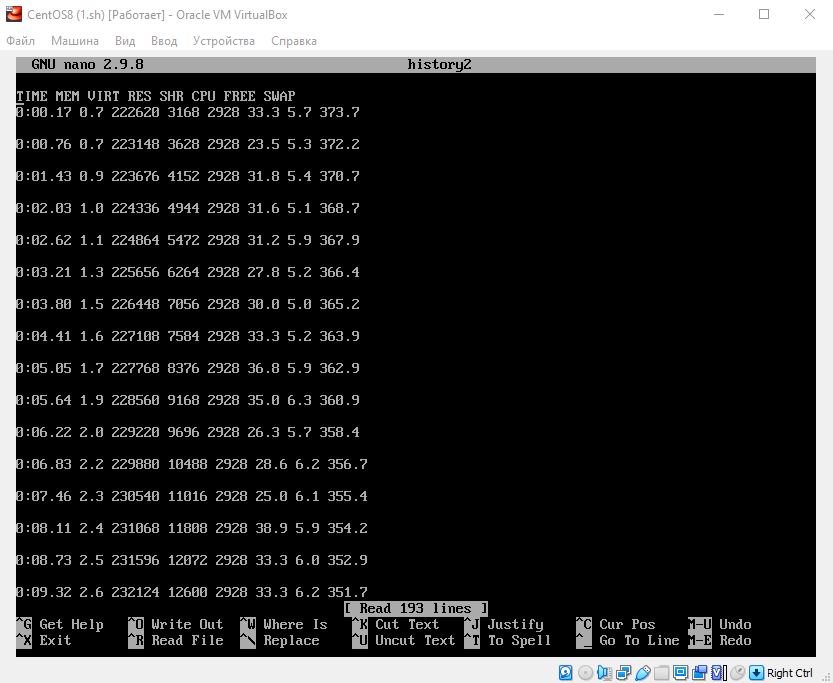
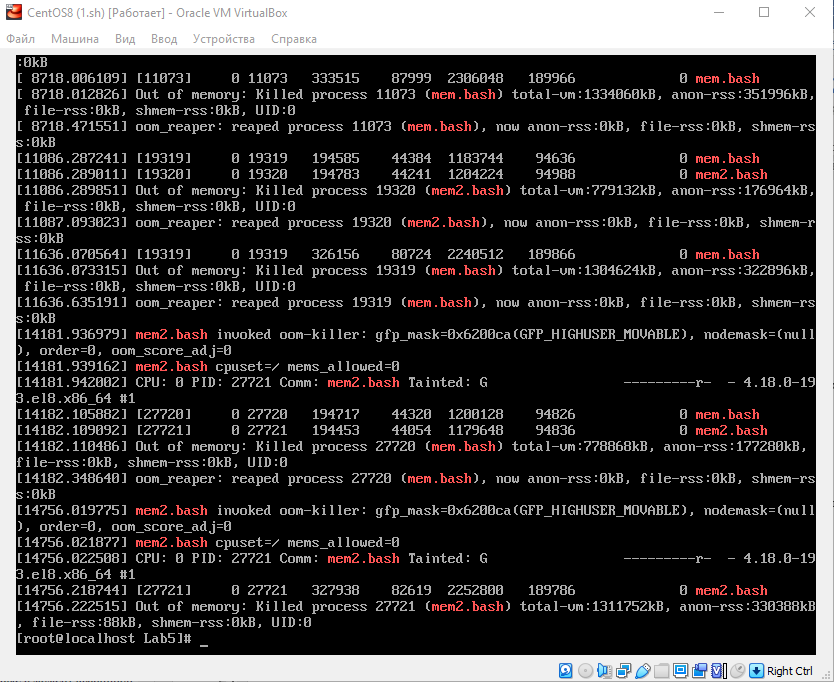
***Подготовительный этап:***

1. Создал скрипт mem.bash, который на каждой итерации цикла добавляет в конец массива последовательность из 10 элементов. И каждый 10000-ный шаг добавлял в файл report.log текущее значение размера массива.

***Первый этап:***

1. Последнее значение в файле report.log: 14220000
2. Данные во время наблюдения из top  
   
3. Записи в системном журнале о аварийном завершении скрипта:  
   
4. График зависимости времени от памяти полученный на основе данных из history  
   

***Второй этап:***

1. Производим аналогичные действия, что и в первом этапе.
2. Данные, полученные во время наблюдения top, для каждого скрипта  
     
   
3. Значение в последние значения из report.log и report2.log: 7120000 c 13940000
4. Записи из системного журнала об аварийном завершении скриптов:  
   
5. Графики зависимости времени от памяти полученный на основе данных:

0

,

00

100,00

200,00

300,00

400,00

500,00

600,00

700,00

800,00

900,00

0:14:24

1:26:24

2:38:24

3:50:24

5:02:24

6:14:24

7:26:24

8:38:24

9:50:24

MEMORY

TIME

Free

Swap

***Вывод из эксперимента №1***

При низкой величине физической памяти (6 – 7 Mб) происходит переход на файлы подкачки. Как только их значение станет критично, то произойдет аварийное завершение программы.

Если запустить 2 скрипта одновременно, то из графика можно заметить, что один из них продолжил работу, это обусловлено тем, что во время аварийного завершения первого скрипта у нас восстанавливается физическая память, далее идет сценарий как из первого этапа.

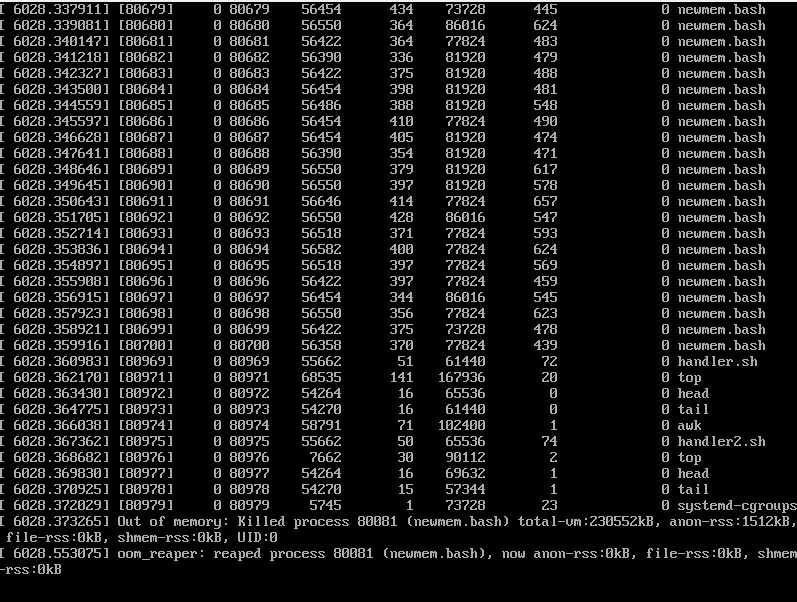
**Эксперимент №2**

1. Скопировал mem.bash под названием newmem.bash и добавил условию на прекращение скрипта
2. Написал скрипт, который будет запускать newmen.bash ежесекундно.
3. Установил K=10 и N=1422000

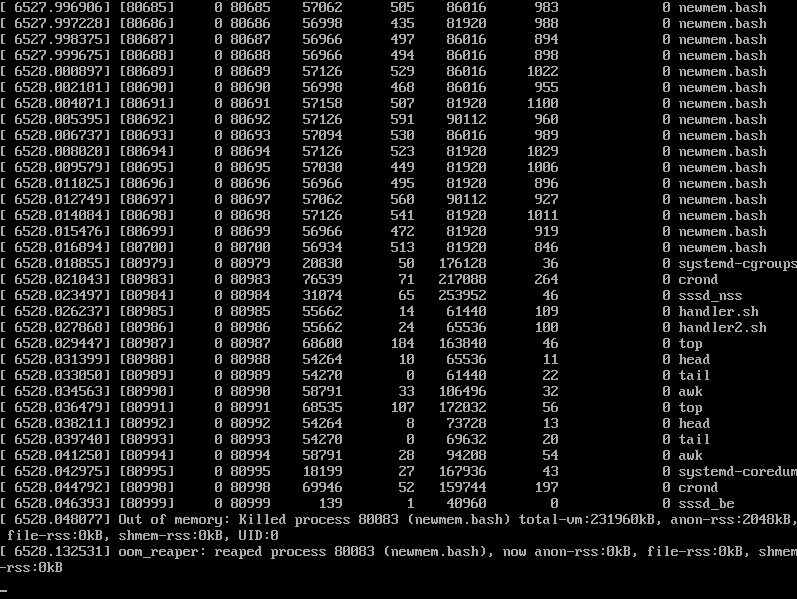
***Выводы по эксперименту №2:***

При К = 10 и N = 1422000программа успешно завершила свое выполнение.

При К = 30 и N = 1422000программа аварийно завершила свое выполнение.



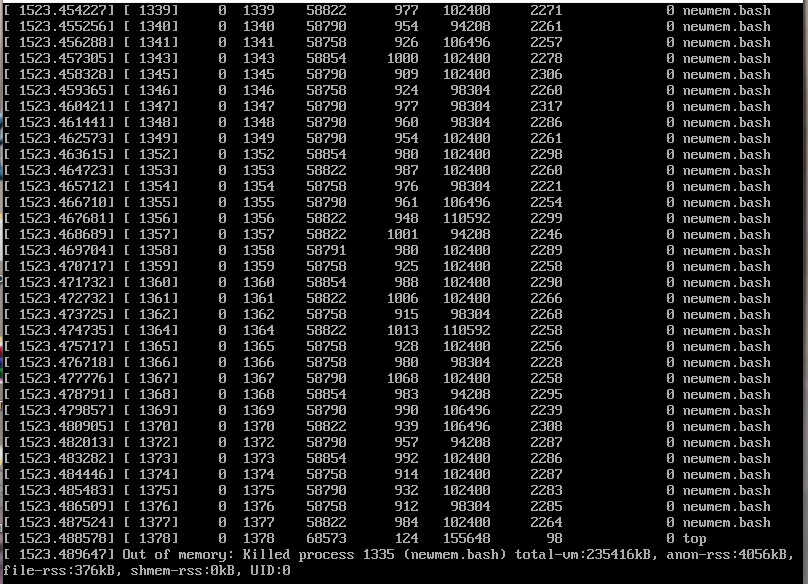
При К = 30 и N = 1300000 программа аварийно завершила свое выполнение.



При К = 30 и N = 1100000 программа успешно завершила свое выполнение.

При К = 30 и N = 1200000 программа успешно завершила свое выполнение**.**

При К = 30 и N = 1250000 программа аварийно завершила свое выполнение.



В итоге мы можем сказать, что для успешного выполнения программы N надо брать из диапазона 1200000-1250000

Все K запусков завершились успешно, это объясняется тем, что первые процессы успевают закончиться раньше, чем наступает критический порог свободной памяти. По мере завершения первых запусков происходит высвобождение памяти для последующих.

**Вывод**

Во время выполнения этой лабораторной работы я понял, что во время работы с оперативной память, данные заполняют физическую память, пока не дойдет до нижнего порога 4-6Мб. Далее ОС начинает заполнять файлы подкачки. В случае, когда файлов уже не хватает, то ОС аварийно завершает процесс.