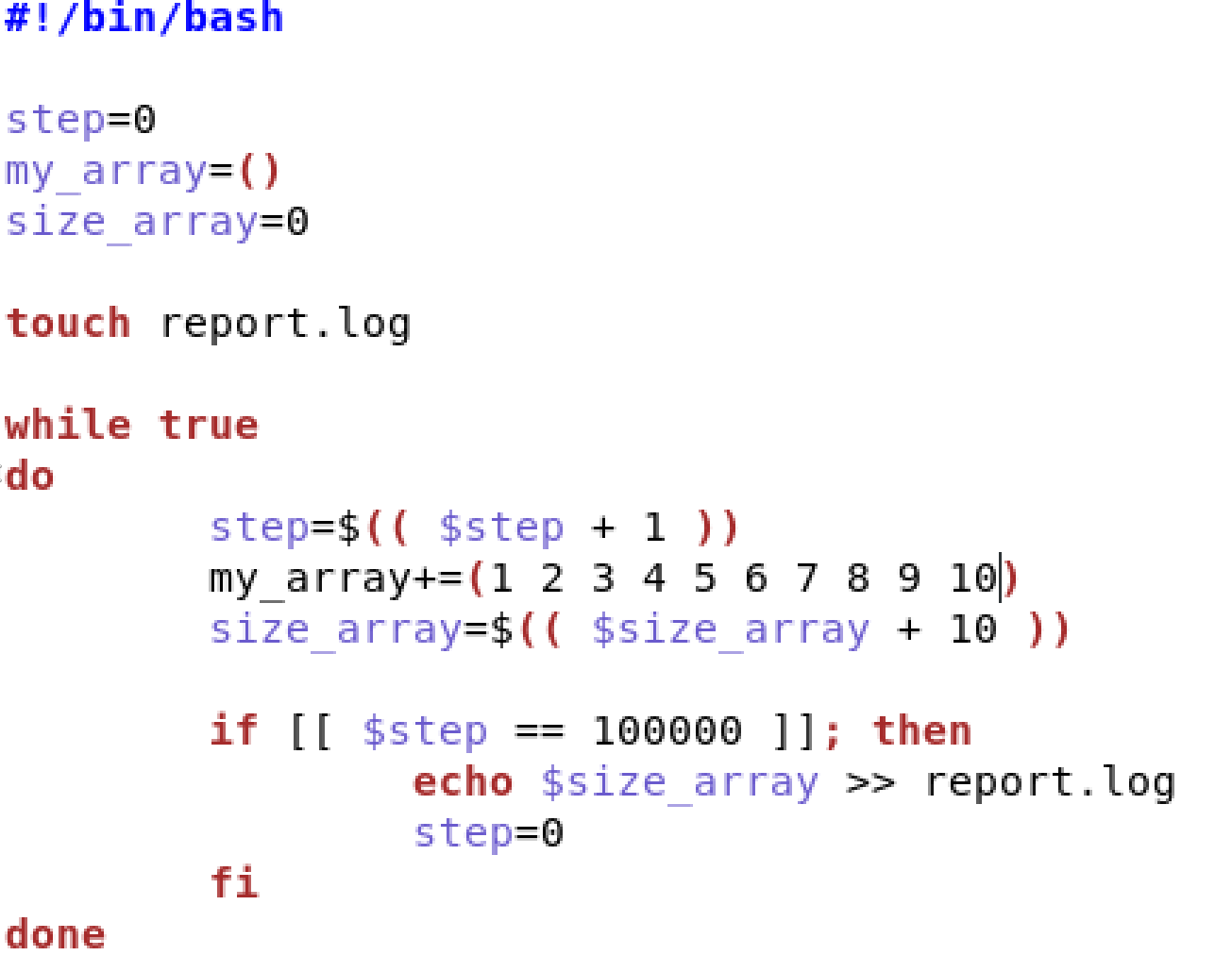
**Лабораторная работа No5. Управление памятью в ОС Linux**

**---Ли Яньюй (М3209)**

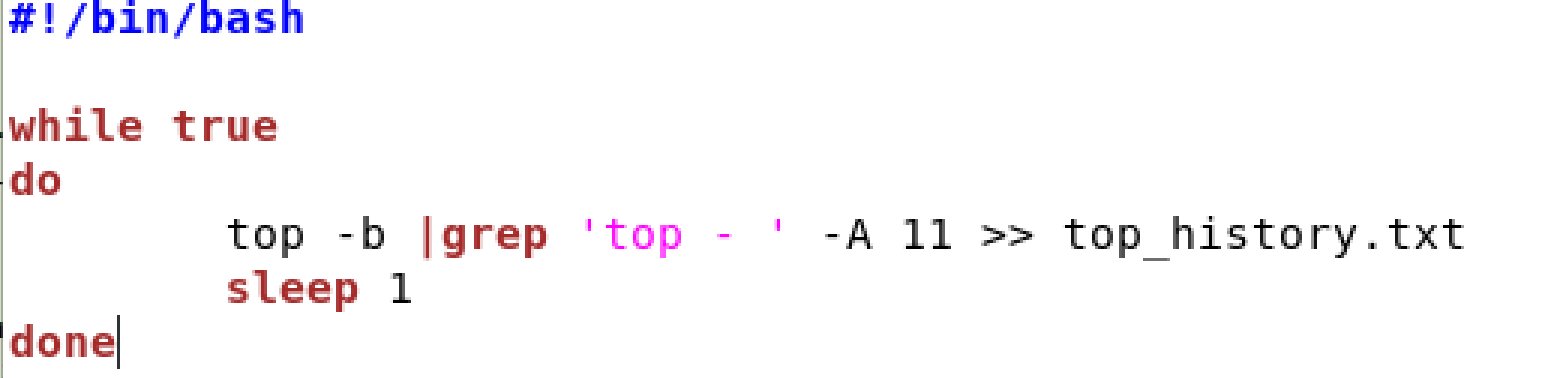
**Эксперимент No1**

**Подготовительный этап:**

1. Создайте скрипт **mem**.

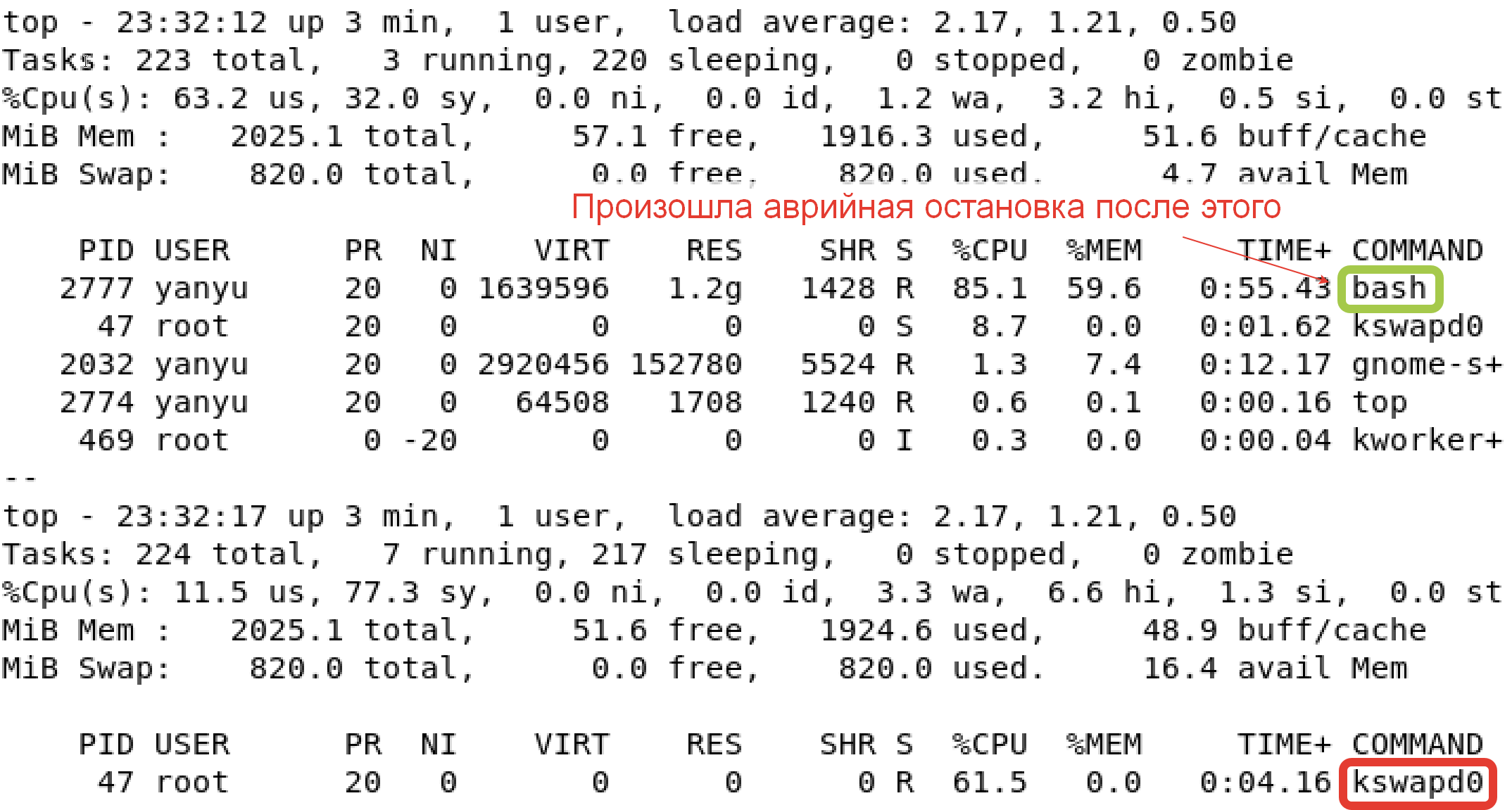


1. Написать «следящие» скрипты и собирать данные, ---- **top\_tracker**

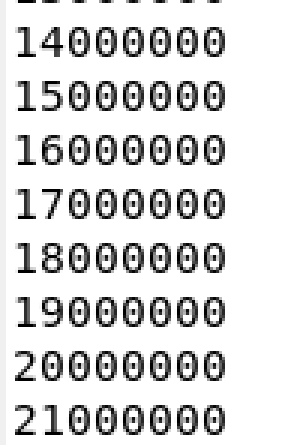


**Первый этап:**

**Последняя запись журнала – значения параметров, с которыми произошла аварийная остановка процесса.**



И **значение в последней строке файла report.log.**



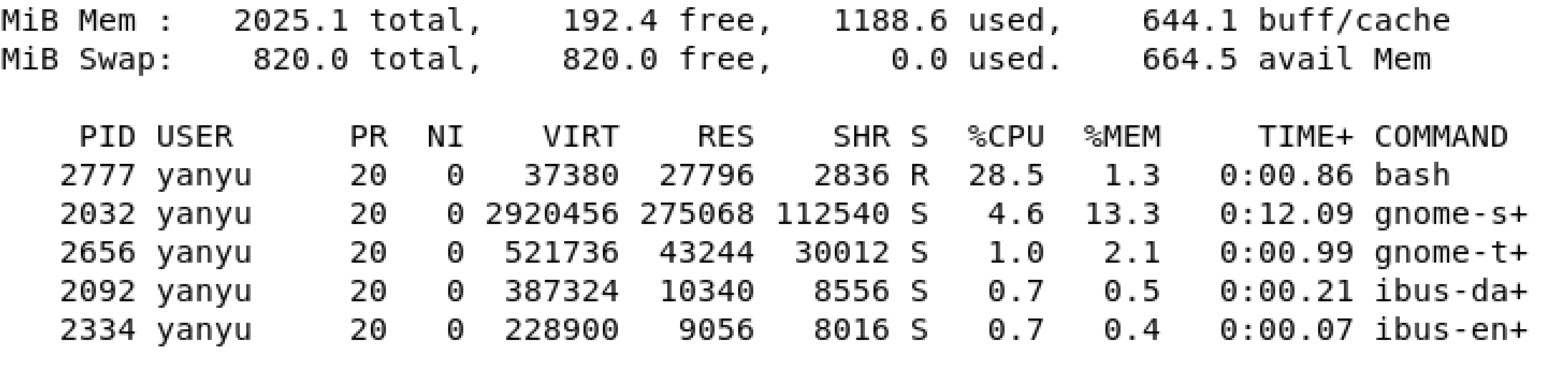
**Наблюдайте за следующими значениями** (и фиксируйте их изменения во времени в отчете):

 1) значения параметров памяти системы (верхние две строки над основной таблицей);

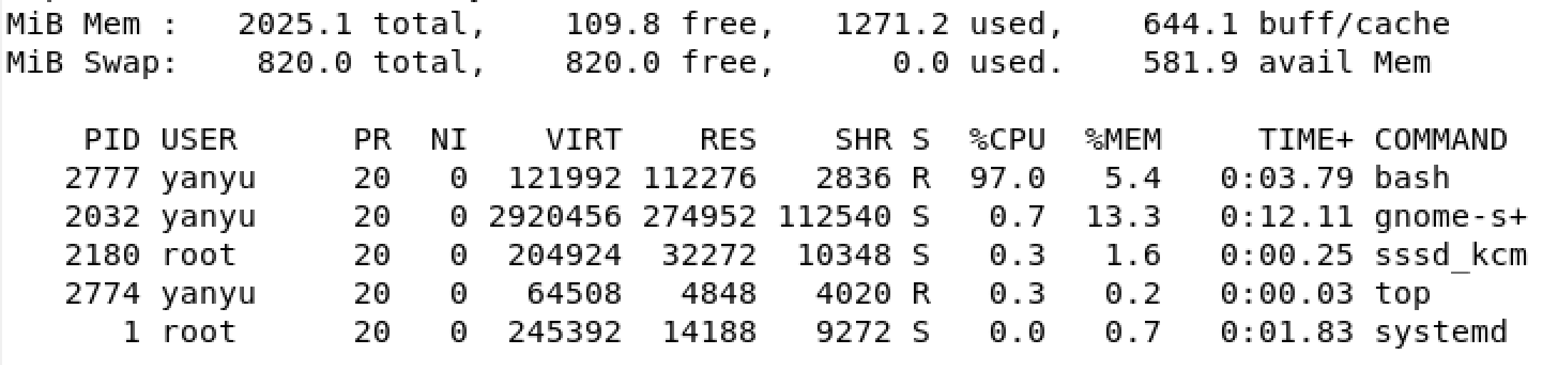
 2) значения параметров в строке таблицы, соответствующей работающему скрипту;

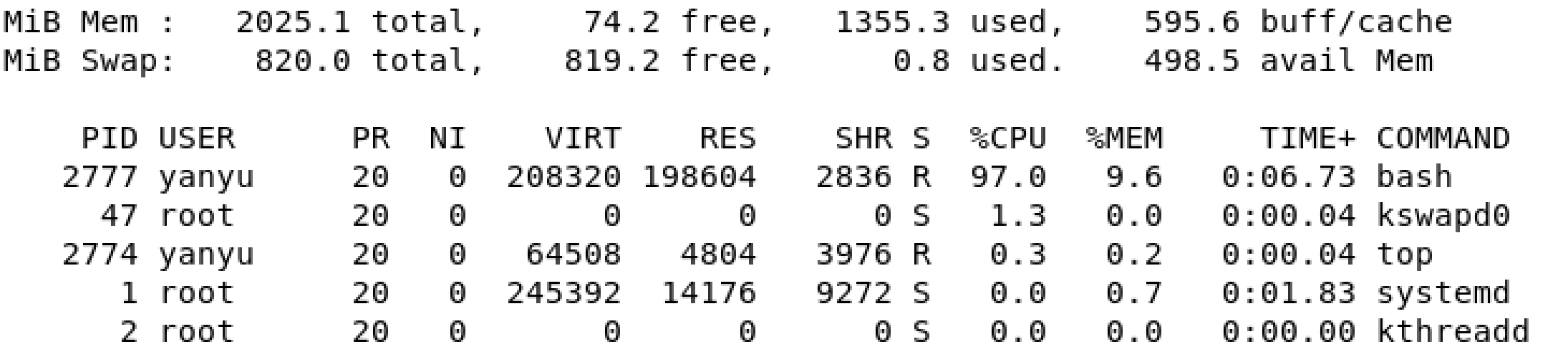
3) изменения в верхних пяти процессах (как меняется состав и позиции этих процессов).

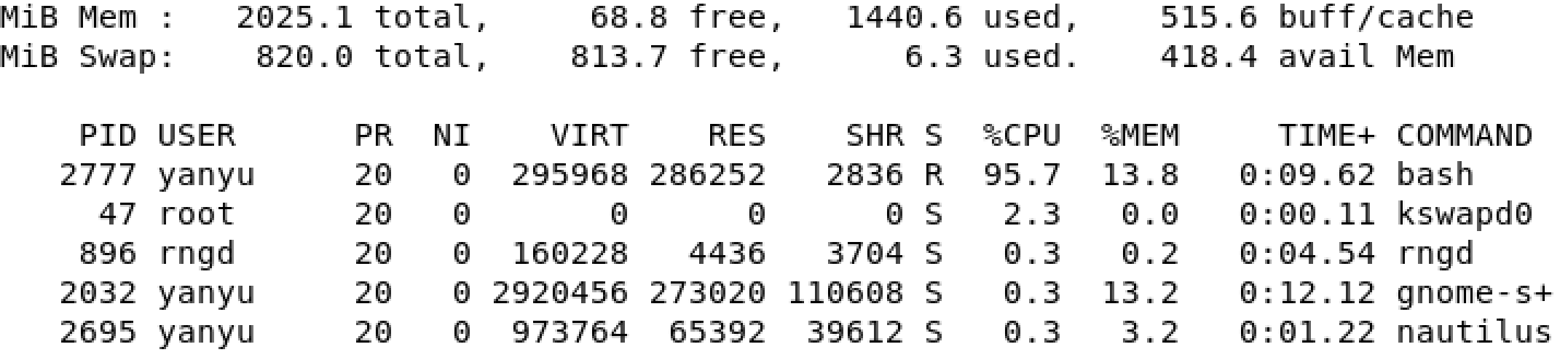
Полные данные находятся в файле **top\_history.txt**

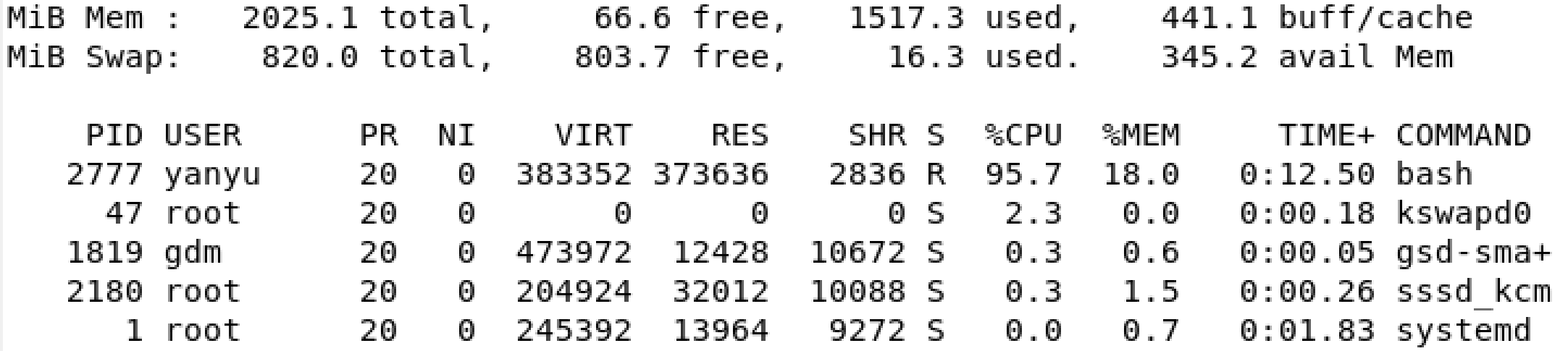


Через почти 1 сек после старта, процесс становится топ1 по требуемой ему оперативной памяти и занимает эту позицию до завершения процесса.



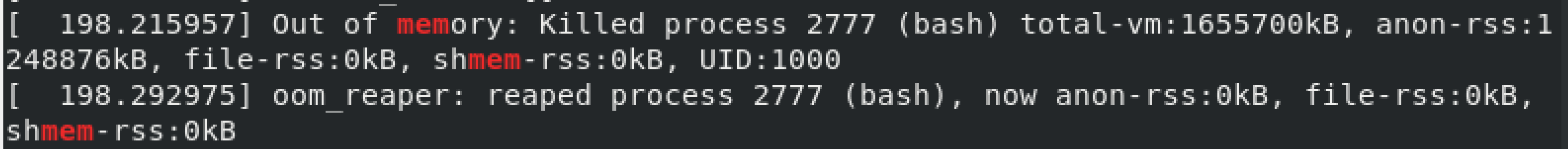






Последние две записи о скрипте в системном журнале

(с помощью команды dmesg | grep "mem")



**Второй этап:**

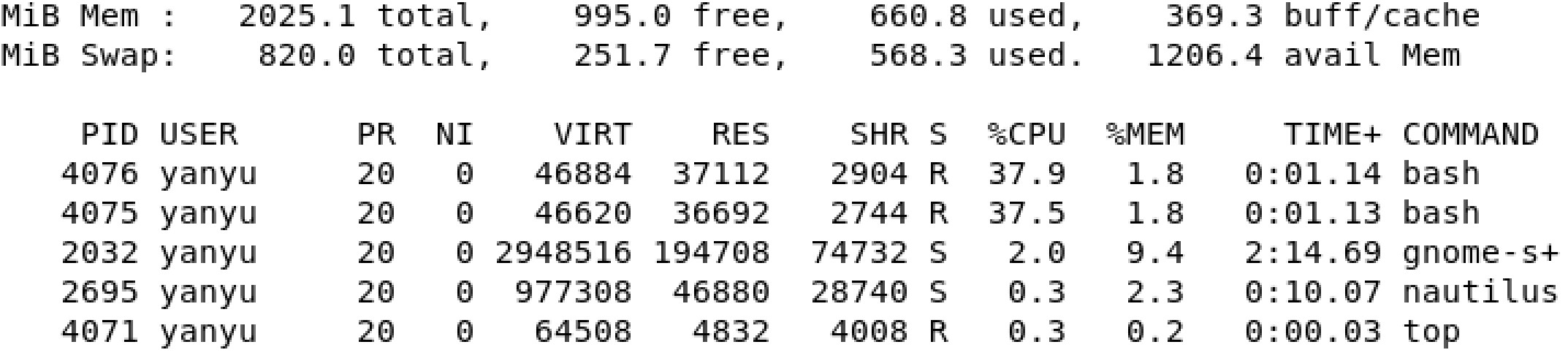
**Наблюдайте за следующими значениями** (и фиксируйте их изменения во времени в отчете):

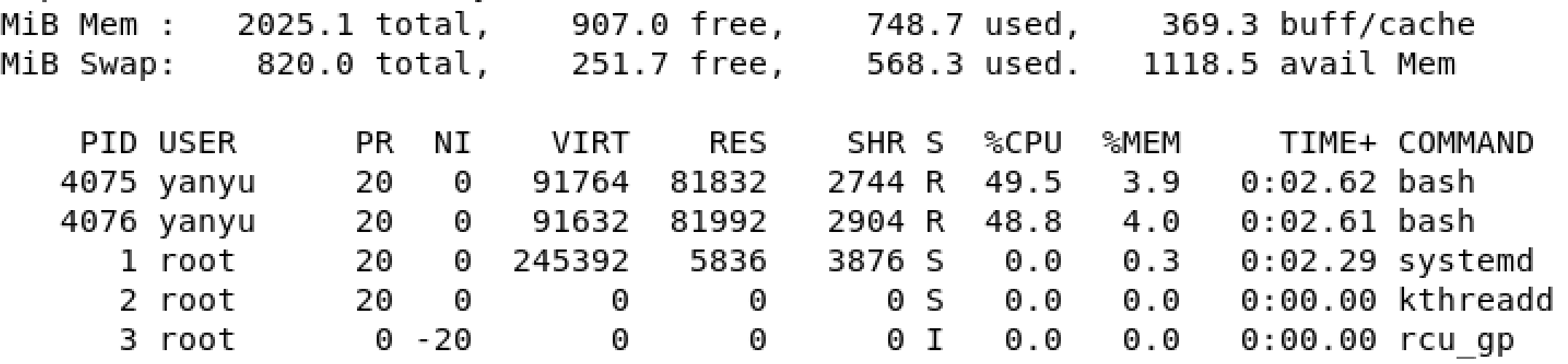
 1) значения параметров памяти системы (верхние две строки над основной таблицей);

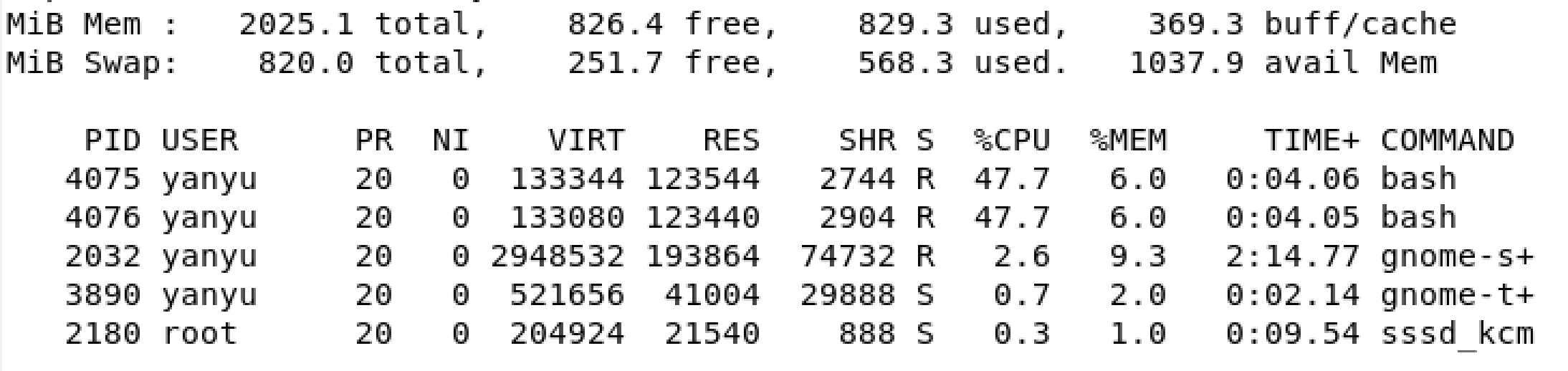
 2) значения параметров в строке таблицы, соответствующей работающему скрипту;

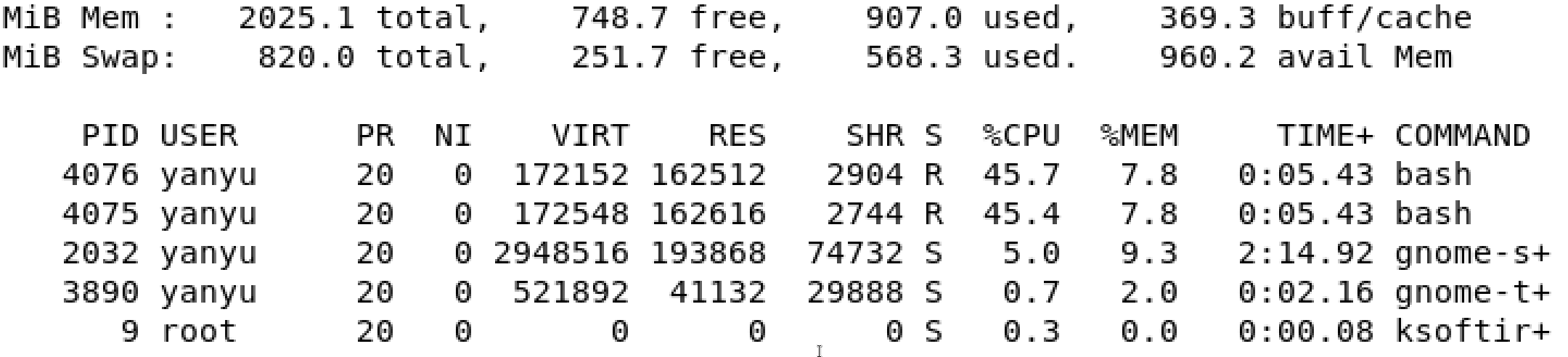
 3) изменения в верхних пяти процессах (как меняется состав и позиции этих процессов).

Полные данные находятся в файле **top\_history.txt**

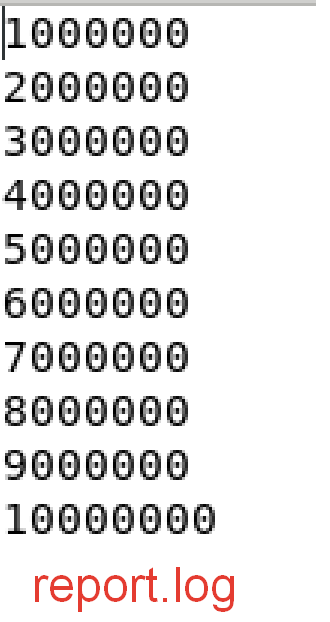
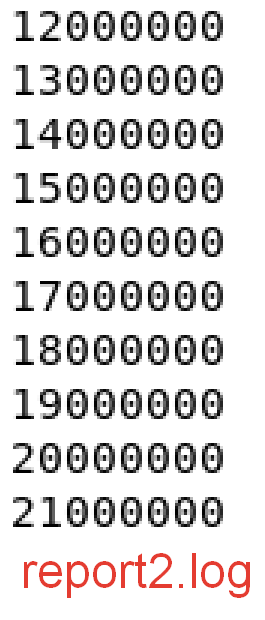






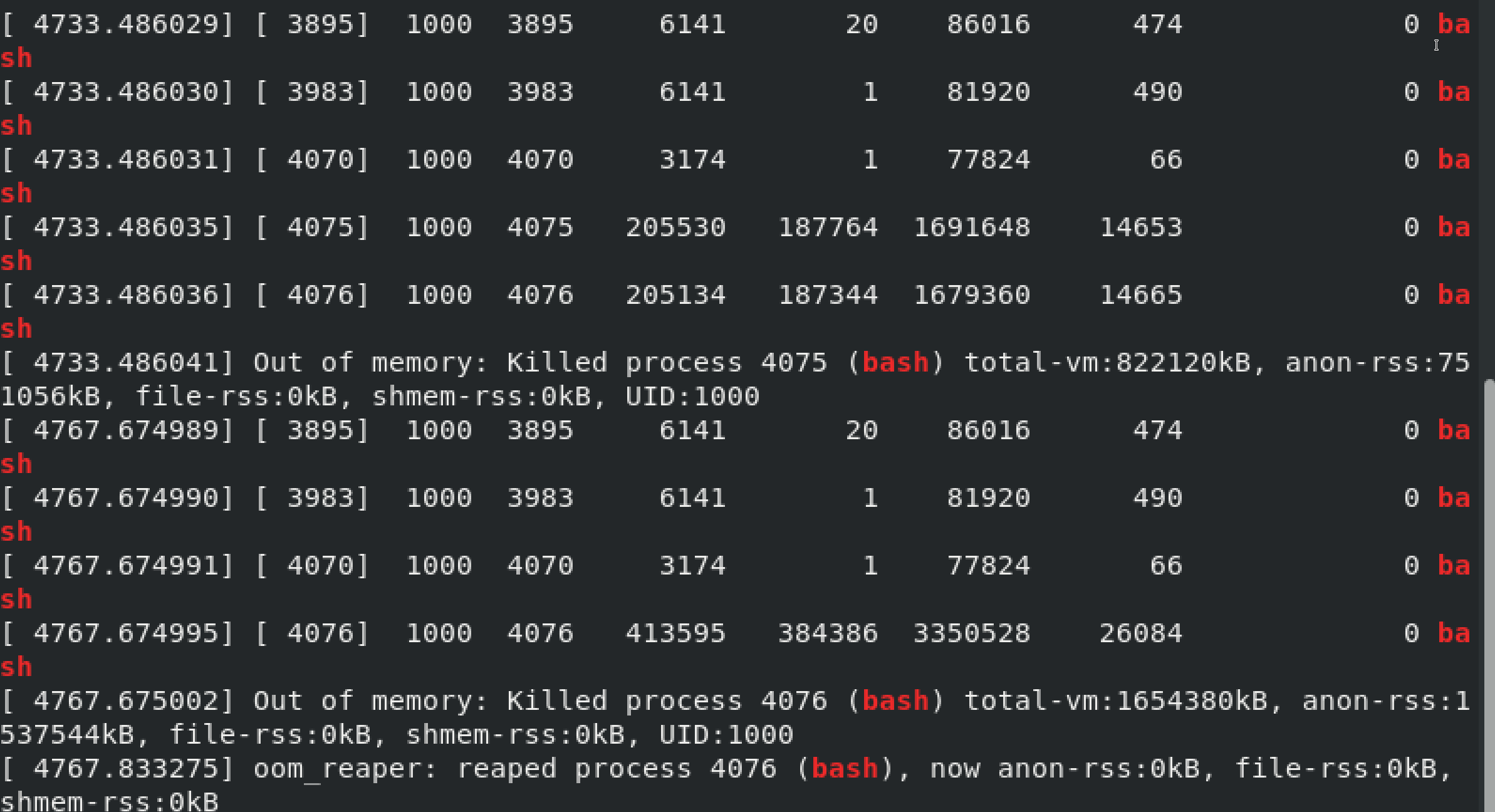


**Значения в последних строках файлов report.log и report2.log.**

**Последние записи о скриптах в системном журнале**

(с помощью команды dmesg | grep "bash[2]\*")

****

**Обработка результатов:**

**Письменные выводы:**

**Первоначально, в оперативной памяти есть место, поэтому процесс пользуется RAM, а swap не пользуется.**

**Потом, когда место оперативной памяти не хватает, происходит копирование страниц из RAM на swap-раздел жёсткого диска.**

**Затем, произошла аварийная остановка процесса, то есть процесс удалён системой, эти страницы удаляются из swap'a и оперативная память освобождается.**

**Письменные выводы:**

**Первоначально, в оперативной памяти есть место, поэтому процесс пользуется RAM. Когда место RAM не хватает, то swap копировались страницы виртуальной памяти. И количество доступной памяти раздела подкачки уменьшалось.**

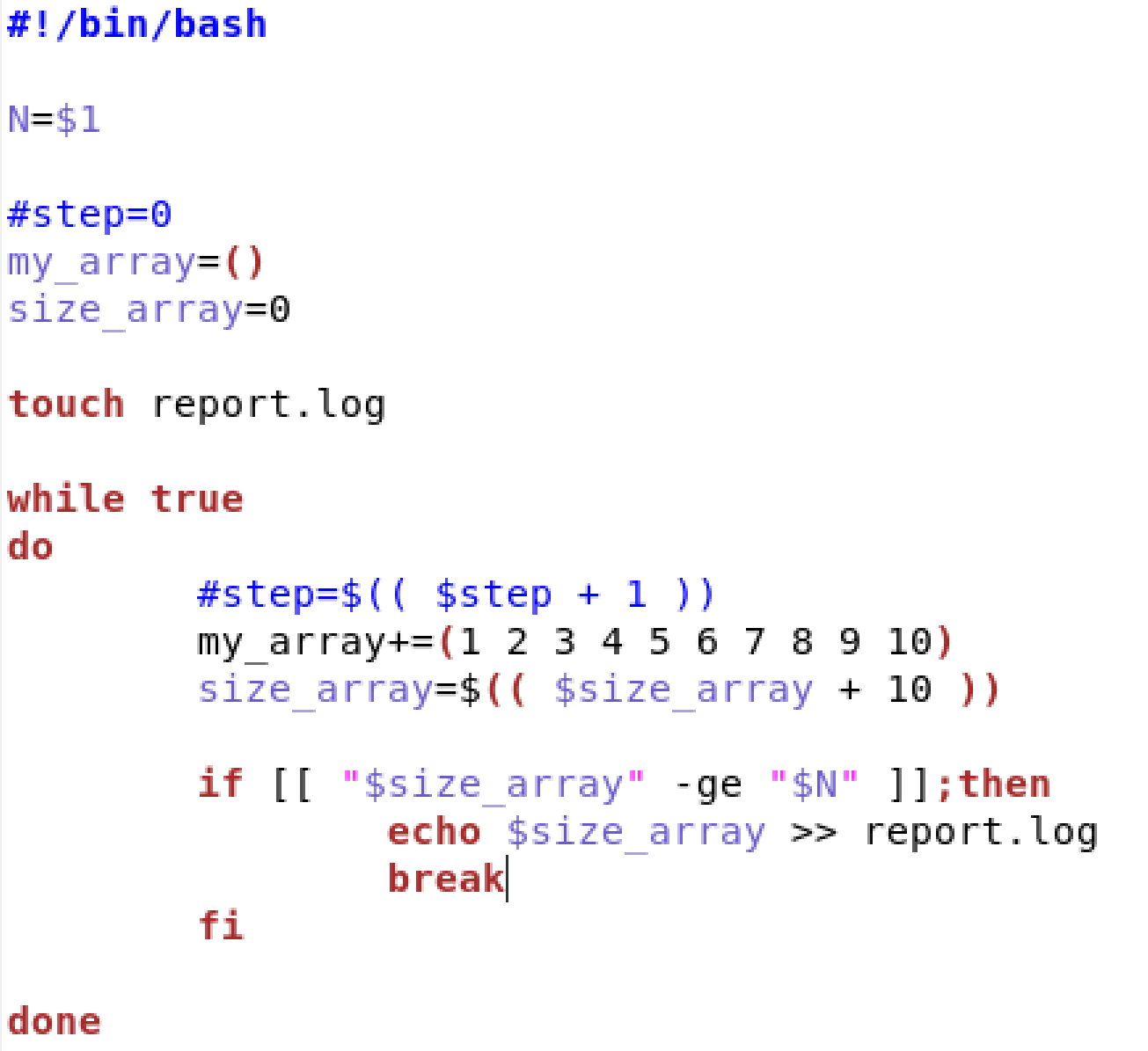
**Затем, произошла аварийная остановка процесса первого скрипта. При этом размеры свободной оперативной памяти и памяти раздела подкачки увеличивались.**

**В оперативной памяти опять есть место, поэтому процесс пользуется RAM……**

**Эксперимент No2**

**Подготовительный этап:**

**Создайте копию скрипта mem.bash в файл newmem.bash. Измените копию таким образом, чтобы она завершала работу, как только размер создаваемого массива превысит значение N, передаваемое в качестве параметра скрипту.**

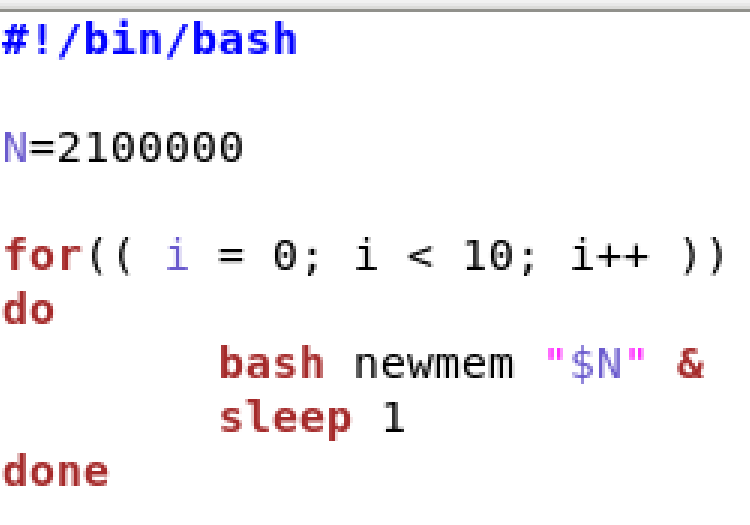
****

**Основной этап:**

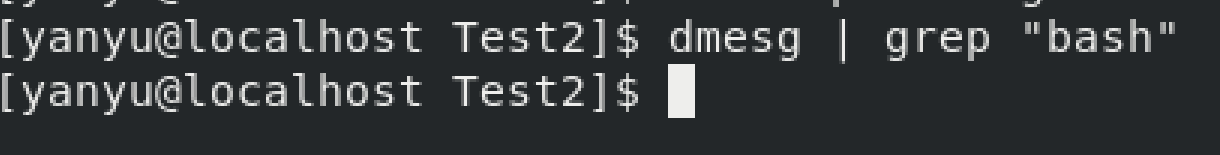
**Скрипт, который будет запускать newmem.bash каждую секунду, используя один и тот же параметр N так, что всего будет осуществлено K запусков.**

**Возьмите в качестве значения N, величину, в 10 раз меньшую, чем размер массива, при котором происходила аварийная остановка процесса в первом этапе предыдущего эксперимента. N = 21000000/10 = 2100000**

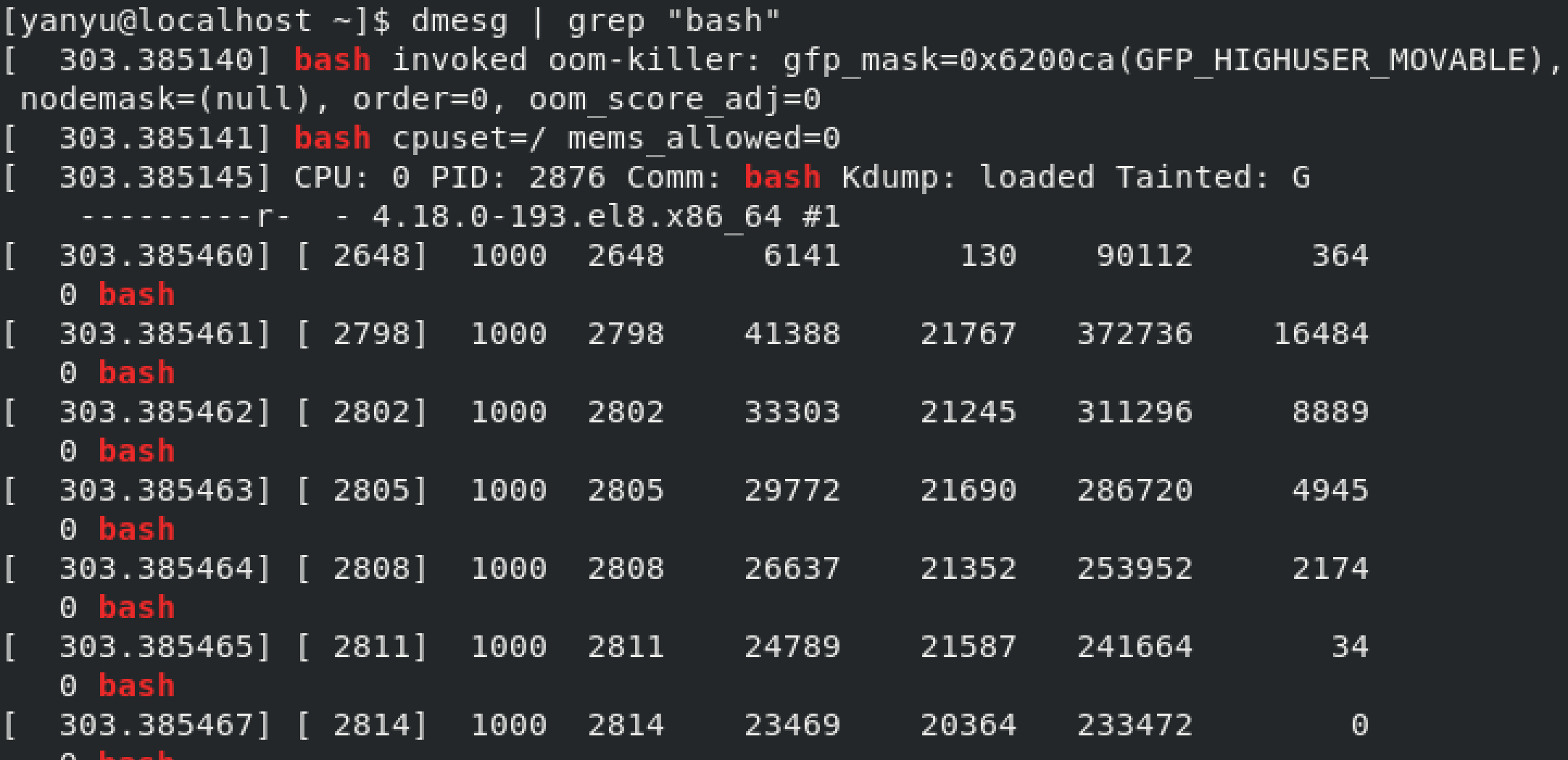
**Возьмите в качестве K значение 10. K = 10**

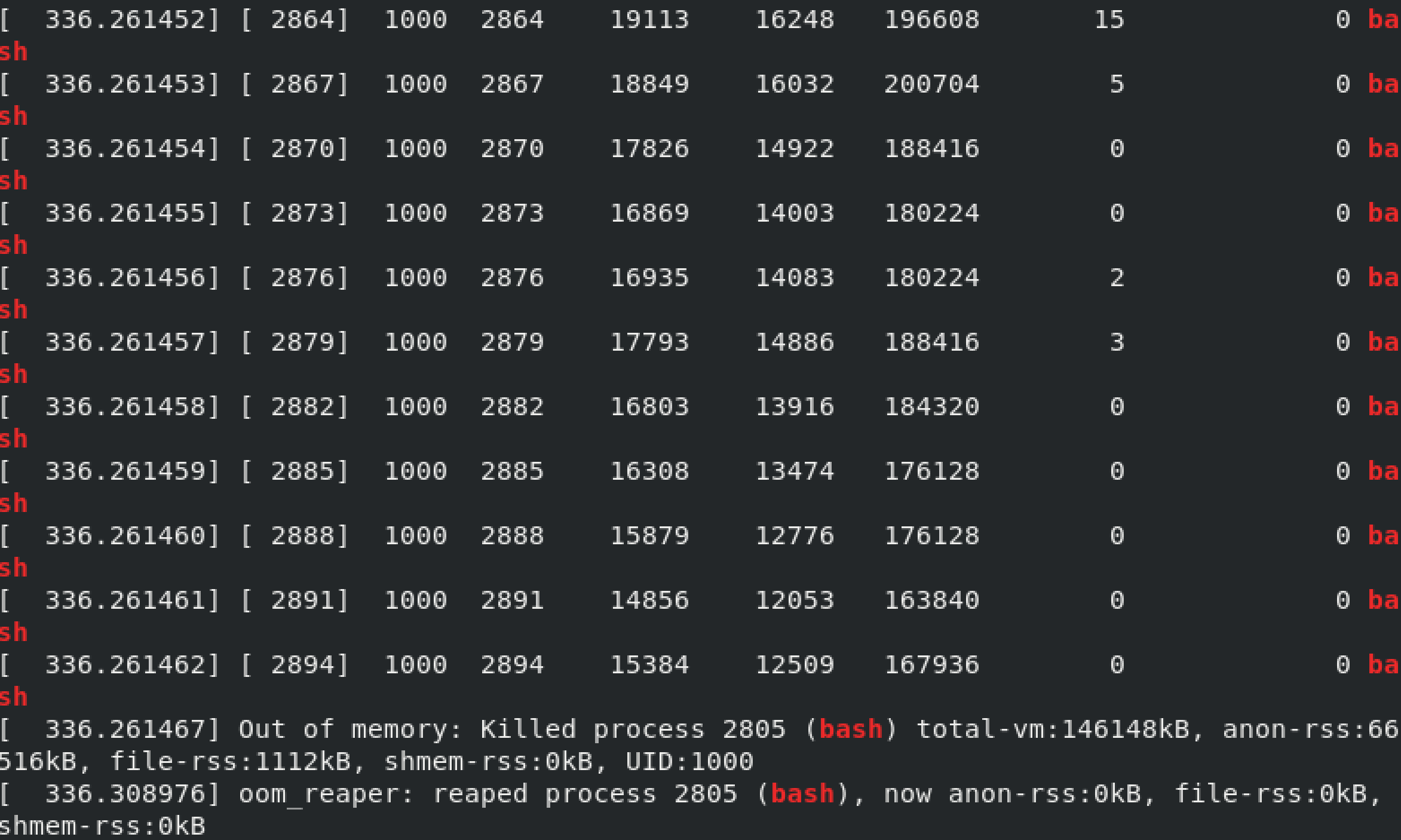
****

**Убедитесь, что все K запусков успешно завершились, и в системном журнале нет записей об аварийной остановке**

****

**Измените значение K на 30 и снова запустите скрипт.**

****

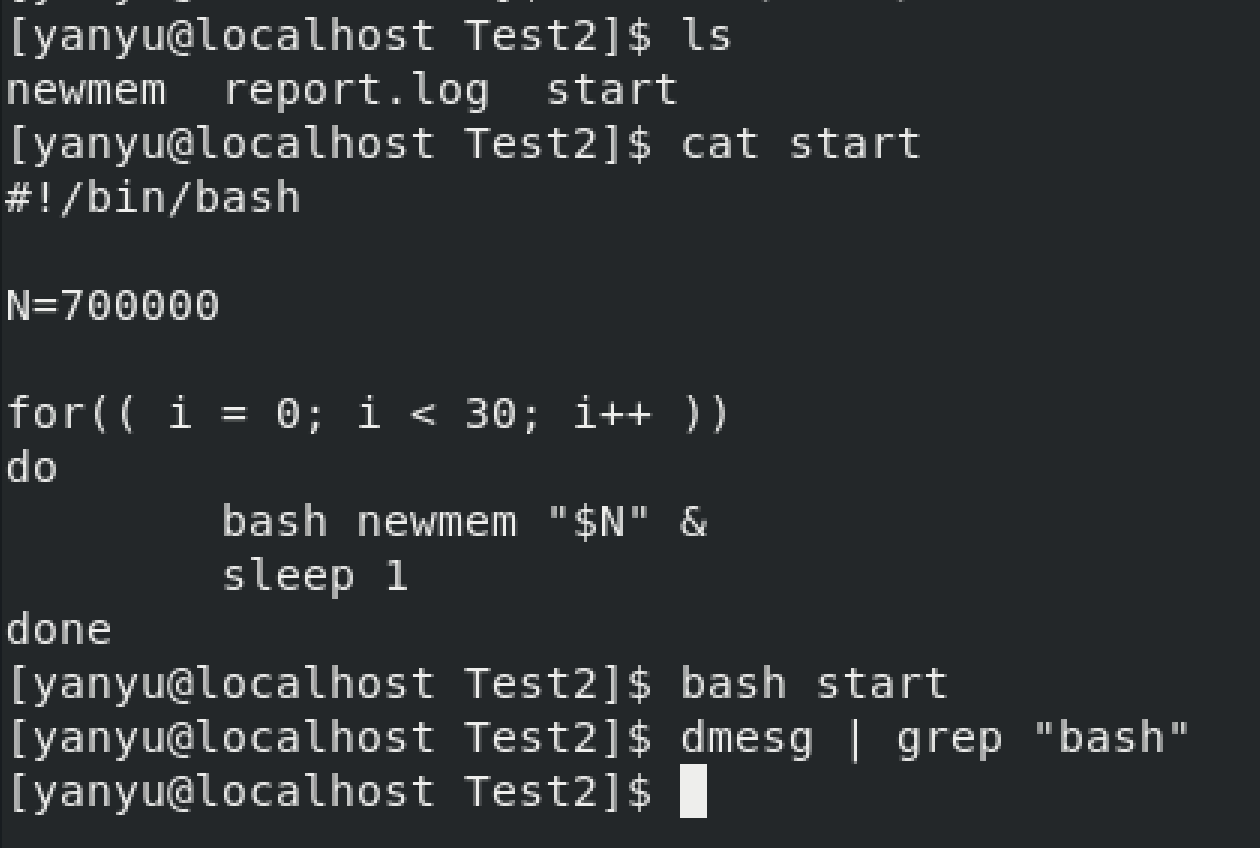
****

**Q:** **Объясните, почему ряд процессов завершился аварийно.**

A: Потому что N \* K > предельное значение, полученное в первом эксперименте.

**Q: Подберите такое максимальное значение N, чтобы при K=30 не происходило аварийных завершений процессов.**

**A:**  N = 2100000/3 = 700000

****