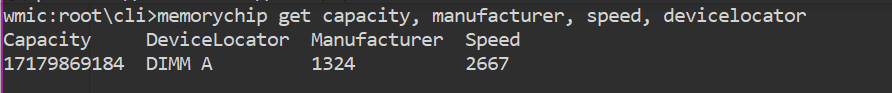
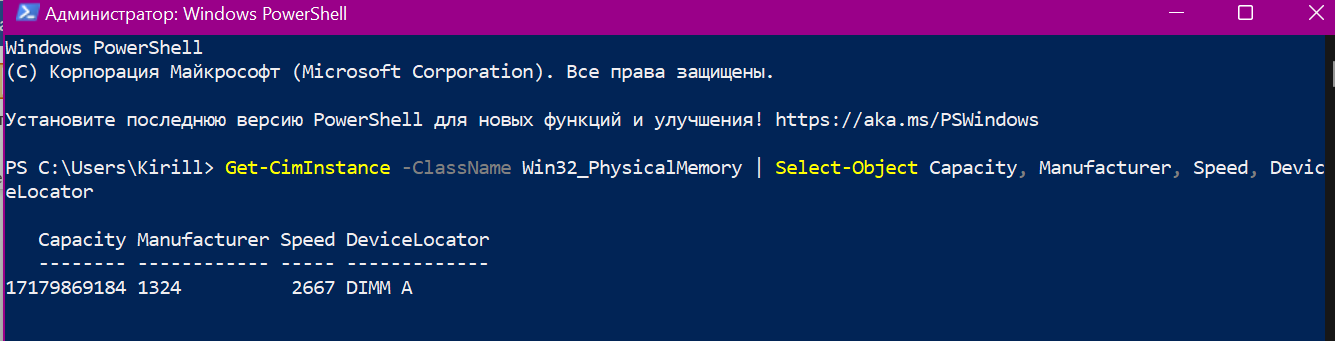
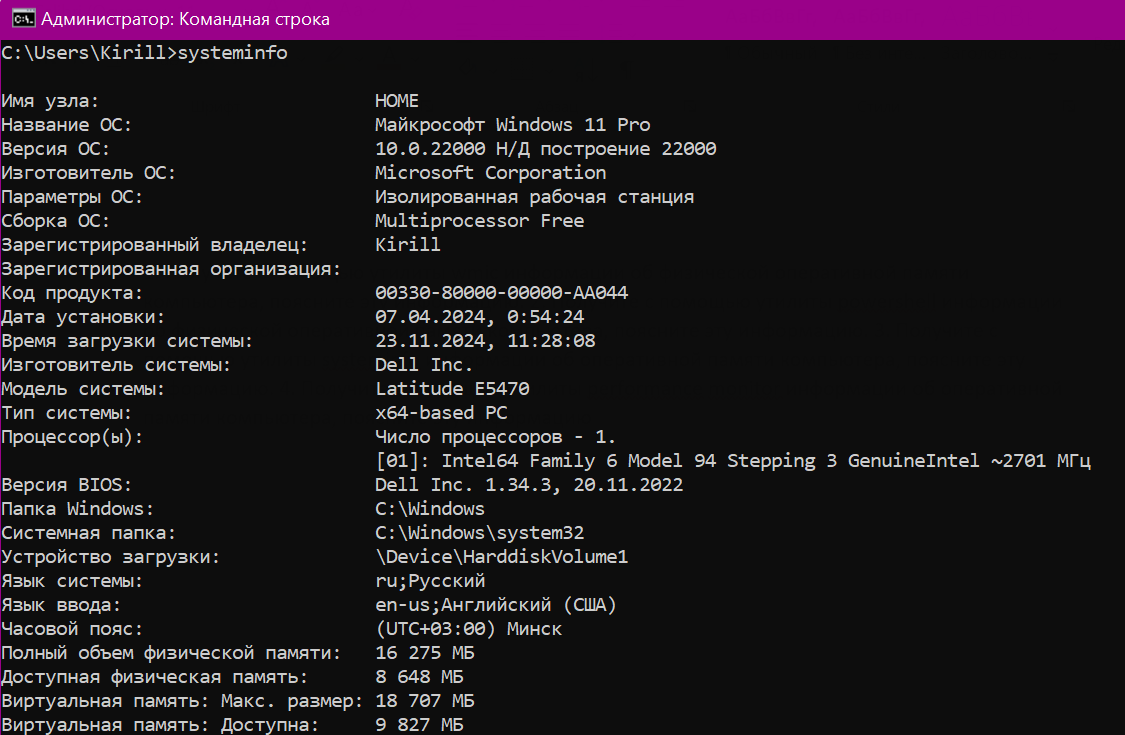
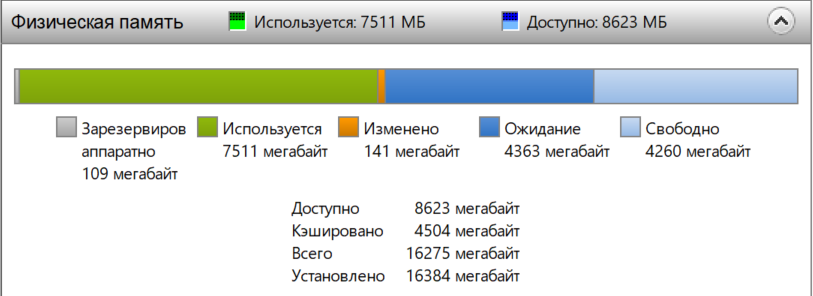
Задание 01. Windows

1. Получите с помощью утилиты wmic информации об физической оперативной памяти компьютера, поясните эту информацию. 2. Получите с помощью утилиты powershell информации об физической оперативной памяти компьютера, поясните эту информацию. 3. Получите с помощью утилиты systeminfo информации об оперативной памяти компьютера, поясните эту информацию. 4. Получите с помощью утилиты performance monitor информации об оперативной памяти компьютера, поясните эту информацию



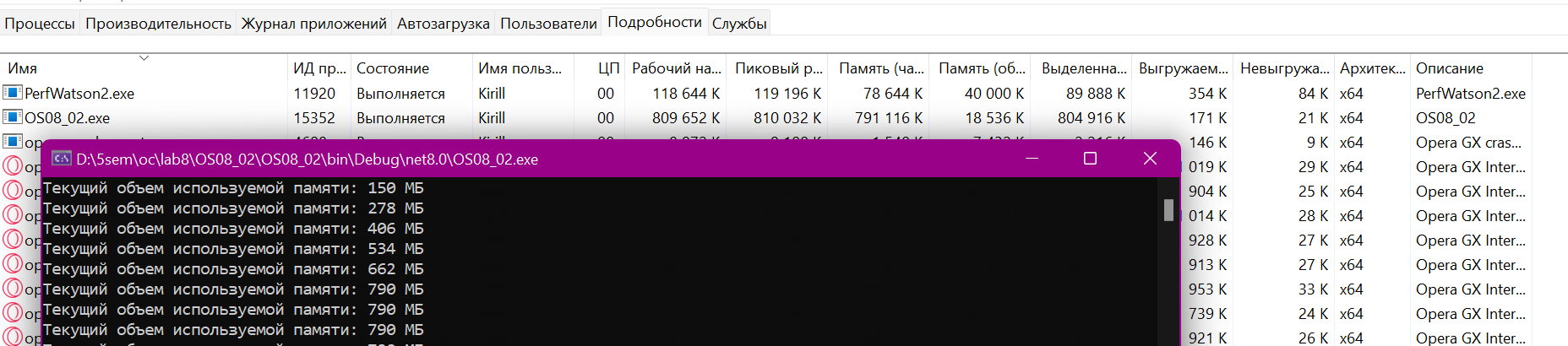


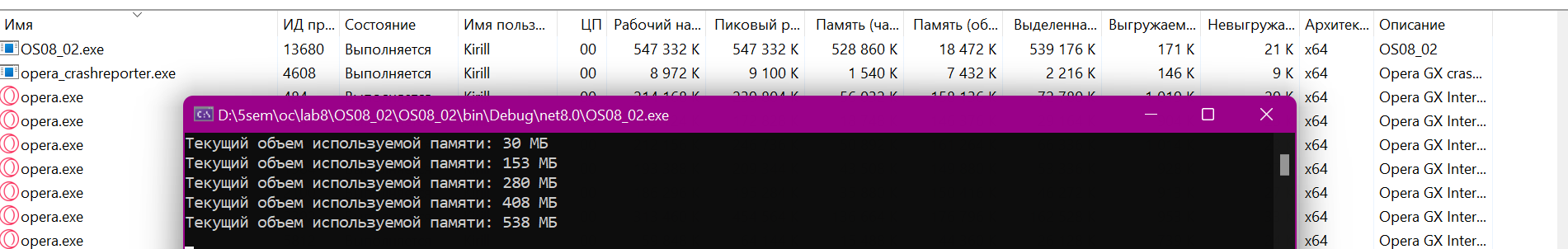




Задание 02.Windows

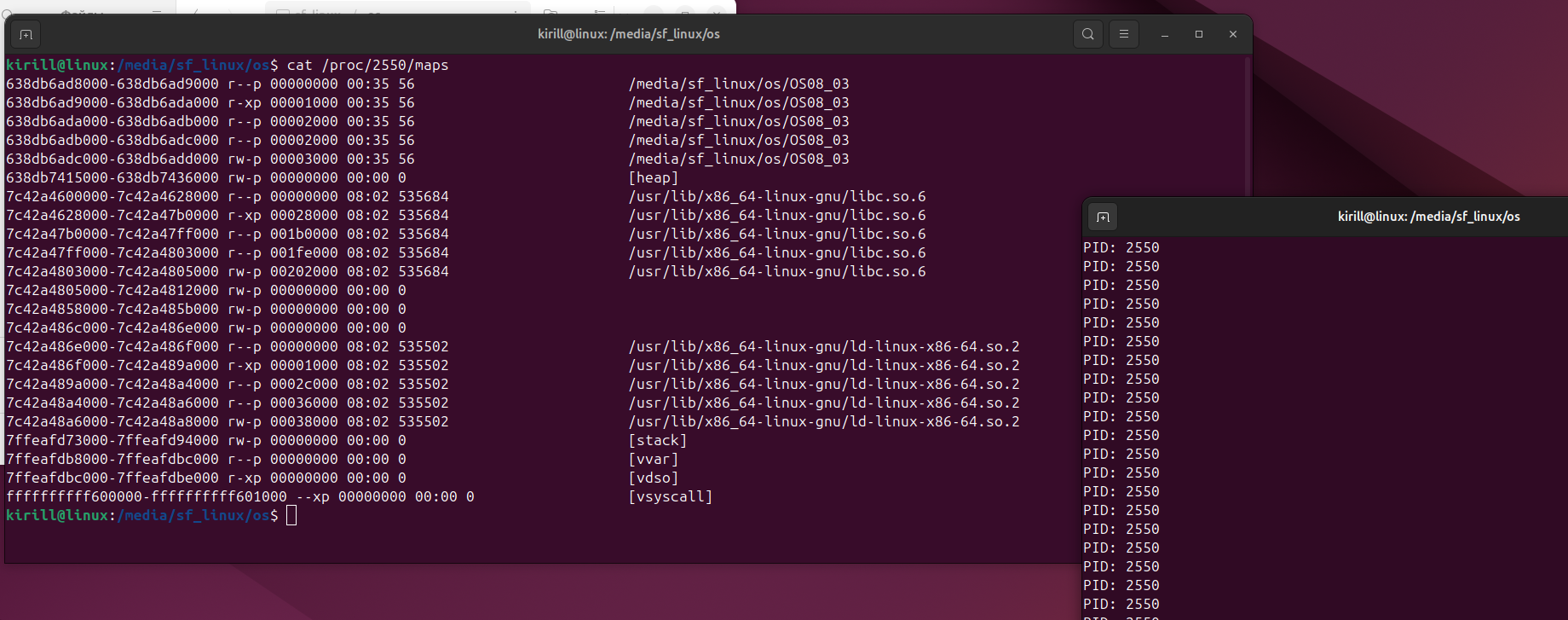
Разработайте на языке программирования C# консольное приложение OS08\_02, которое каждые 5 секунд создает новый объект размером 128 МБ и выводит объем используемой памяти, установите Platform Target x86.

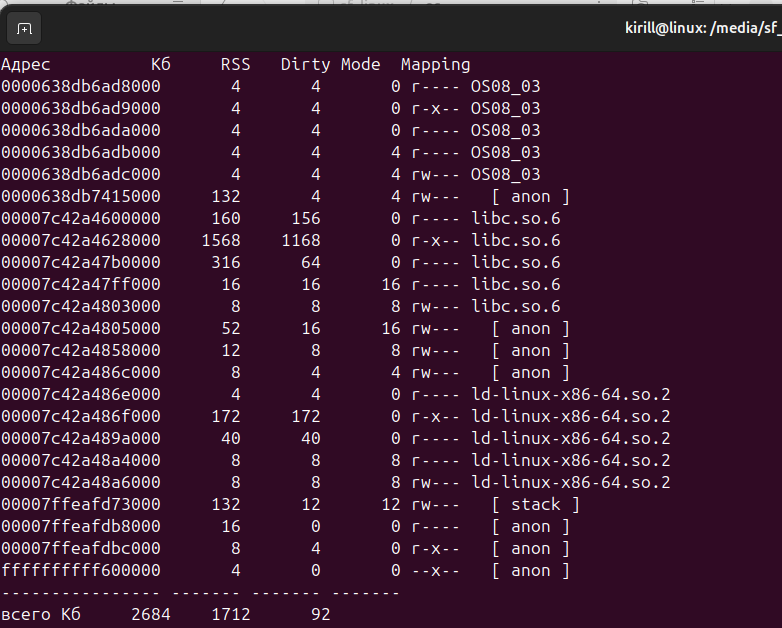


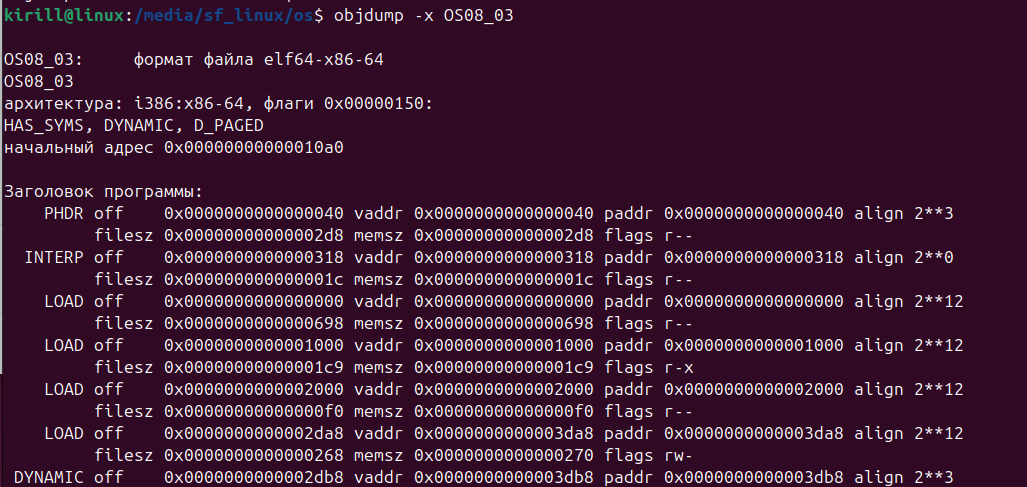


Задание 03.

Linux 9. Разработайте консольное приложение OS08\_03, выполняющее длинный цикл. 10. Продемонстрируйте с помощью файловой системы /proc структуру адресного пространства. 11. Продемонстрируйте с помощью pmap структуру адресного пространства. 12. Определите с помощью утилиты objdump адрес загрузки main модуля, секций с кодом, данными, неинициализированными глобальными переменными.







Укажите содержимое байта по виртуальному адресу 0x01579333, если регистр CR3 содержит значение 0x13907000.

Запишите виртуальный адрес в двоичном виде: 0000 0001 0101 0111 1001 0011 0011 0011

0x01579333 → 0000 0001 0101 0111 1001 0011 0011 0011 (32 бита).

***индекс каталога страниц***: 0000 0001 01 = 0x5 (в десятичной: 5)

Индекс каталога страниц: первые 10 бит → 0000 0001 01

01 0111 1001

***индекс таблицы страниц***: 01 0111 1001= 0x179 (в десятичной: 377)

Индекс таблицы страниц: следующие 10 бит → 01 0111 1001

***Базовый адрес физической страницы памяти: 0x*13908127**

Физический адрес записи в каталоге страниц = 0x13907000+(0x5 \* 4)=0x13907014

базовый адрес таблицы страниц = 13908000

**Физический адрес записи таблицы страниц** = базовый адрес таблицы страниц + (индекс таблицы страниц \* 4).

(0x179\* 4) = 5E4

Физический адрес записи таблицы = 0x13908000 + (0x179\* 4) = 139085E4.

базовый адрес физической страницы = 0x25024000

Смещение: последние 12 бит → 1100 1100 1100 (0x333 в шестнадцатеричной системе).

**Физический адрес** = базовый адрес физической страницы памяти + смещение.

***Искомый физический адрес***: 0x25024333

***Требуемые данные: 0x*75BF6DD3**

Задание 05 Ответьте на следующие вопросы

14. Поясните понятие «виртуальная память».

это механизм, который позволяет операционной системе использовать физическую память (RAM) более эффективно, предоставляя программам адресное пространство, которое может превышать реальный объем доступной памяти.

15. Поясните понятие «свопинг».

Файл подкачки или виртуальная память — это способ системы виртуальной памяти увеличить оперативную память, когда ее не хватает для совершения операций. Система автоматически задействует файл подкачки, когда приложениям не хватит системной памяти ОЗУ. Хотя система сама регулирует объем файла подкачки иногда может понадобиться вручную увеличить виртуальную память.

16. Поясните понятие «страничная память».

Страничная память — способ организации виртуальной памяти, при котором виртуальные адреса отображаются на физические постранично.

Виртуальная память делится на страницы. Размер размера страницы задается процессором и обычно на x86-64 составляет 4 KiB. Это означает, что управление памятью в ядре выполняется с точностью до страницы. Когда вам понадобится новая память, ядро предоставит вам одну или несколько страниц.

17. Поясните понятие MMU.

Ставится задача транслировать 48-битный виртуальный адрес в физический. Она решается аппаратным обеспечением — блоком управления памятью (memory management unit, MMU). Этот блок является частью процессора. Чтобы транслировать адреса, он использует структуры данных в оперативной памяти, называемые таблицами страниц.

устройство, которое управляет виртуальной памятью, выполняя преобразование виртуальных адресов в физические и управляя доступом к памяти

18. Поясните понятие TLB.

кэш, используемый MMU для хранения недавно использованных преобразований виртуальных адресов в физические, что ускоряет доступ к памяти.

19. Какая информация содержится в строке таблицы страниц

содержит информацию о физическом адресе страницы в памяти, атрибуты доступа (например, чтение/запись), флаги (например, наличие страницы в памяти или на диске) и возможные другие метаданные.

20. Поясните принцип применения хэш-таблиц.

применяются для эффективного поиска данных с помощью хэш-функции. Ключи данных преобразуются в индекс, где хранится значение. Это ускоряет операции поиска, вставки и удаления.

21. Поясните применение «инвертированной таблицы физических» страничной памяти.

В отличие от традиционных таблиц страниц, где каждая запись соответствует виртуальной странице, здесь каждая запись соответствует физической.

22. Windows: перечислите области адресного пространства (от младших к старшим адресам) и поясните их назначения.

 **0x00000000 - 0x7FFFFFFF** — пространство процесса (память пользователя).

 **0x80000000 - 0xFFFFFFFF** — пространство ядра (системные ресурсы).

23. Windows: какой стандартный начальный размер области heap?

24. Windows: какой стандартный размер области памяти stack?

25. Windows: что такое «рабочее множество»? поясните принцип управления рабочим множеством с помощью OS API.

это набор страниц памяти, которые активно используются процессом в данный момент. OS API управляет рабочим множеством, перемещая страницы в память или выгружая их в файл подкачки.

26. Windows: что такое «heap»? Что такое «heap процесса»? Что такое «пользовательская heap»? Поясните принцип устройства heap. з

Куча — это сегмент памяти, для которого не устанавливается постоянный размер перед компиляцией и который может динамически управляться программистом

 **Heap процесса** — это участок памяти, выделенный для нужд процесса.

 **Пользовательская heap** — часть heap, которая может быть использована программами для выделения памяти с учетом их специфических потребностей.

27. Linux: перечислите области адресного пространства (от младших к старшим адресам) и поясните их назначения.

У каждого процесса в системе Linux есть адресное пространство, состоящее из трех логических сегментов: текста, данных и стека

Текстовый сегмент (text segment) содержит машинные команды, образующие исполняемый код программы.

Сегмент данных (data segment) содержит переменные, строки, массивы и другие данные программы