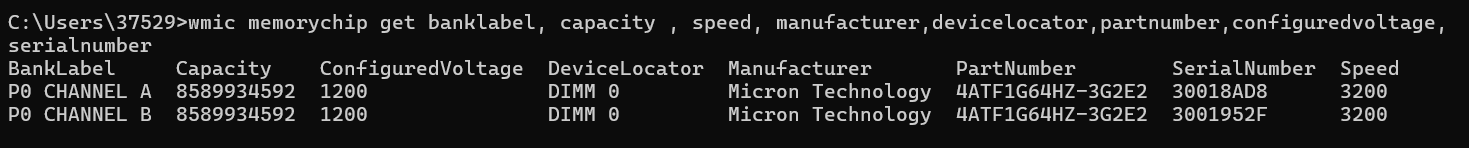
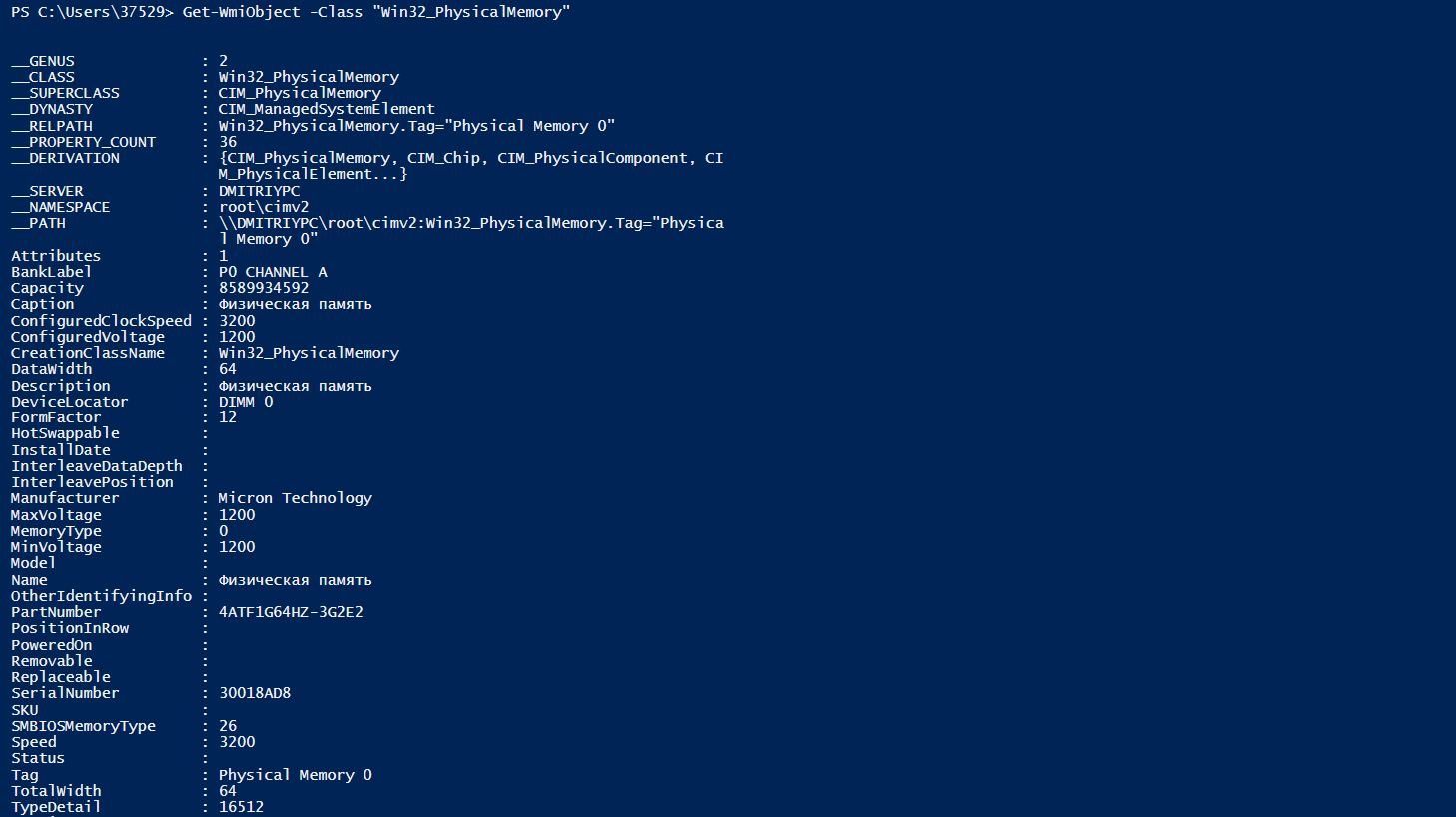
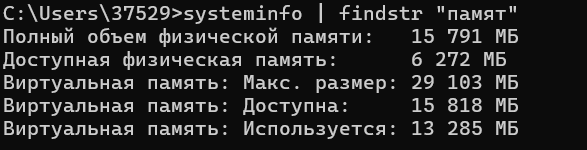
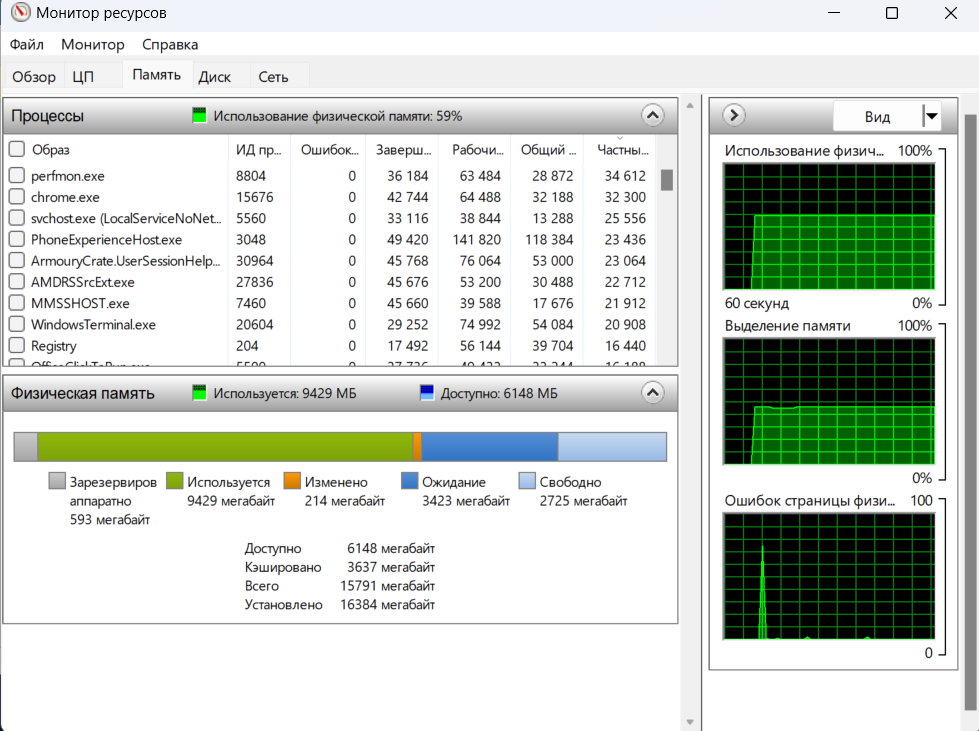
**Задание 01. Windows**

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.



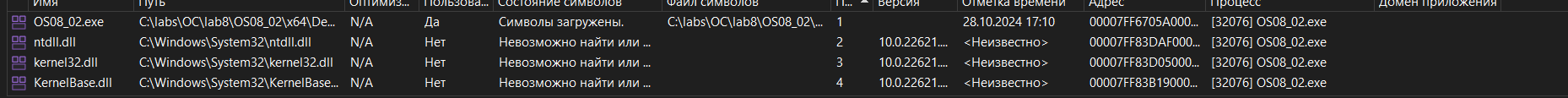


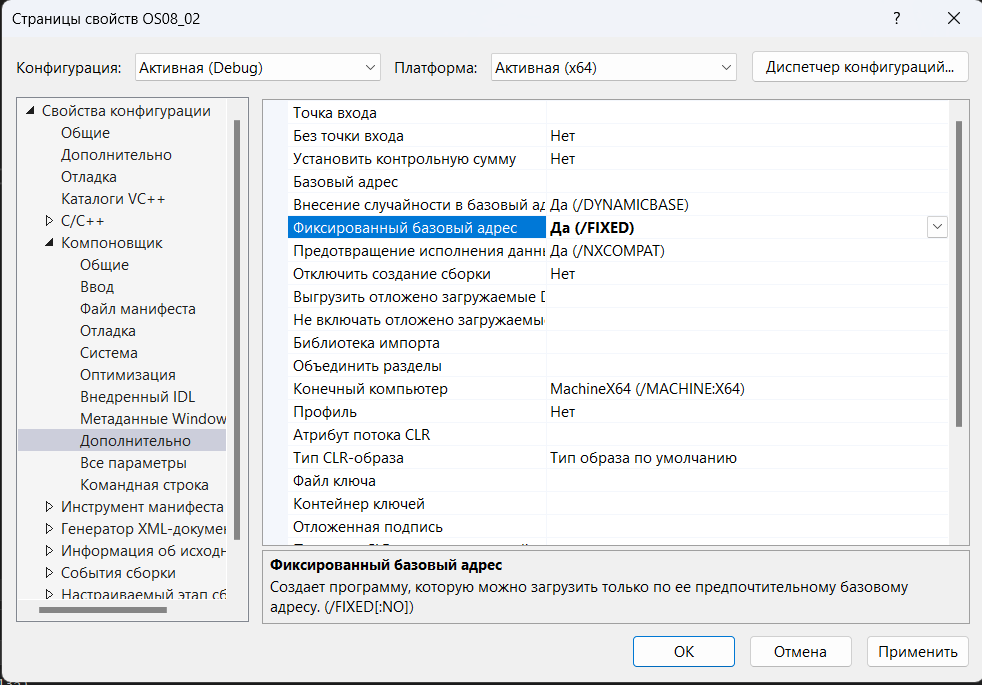


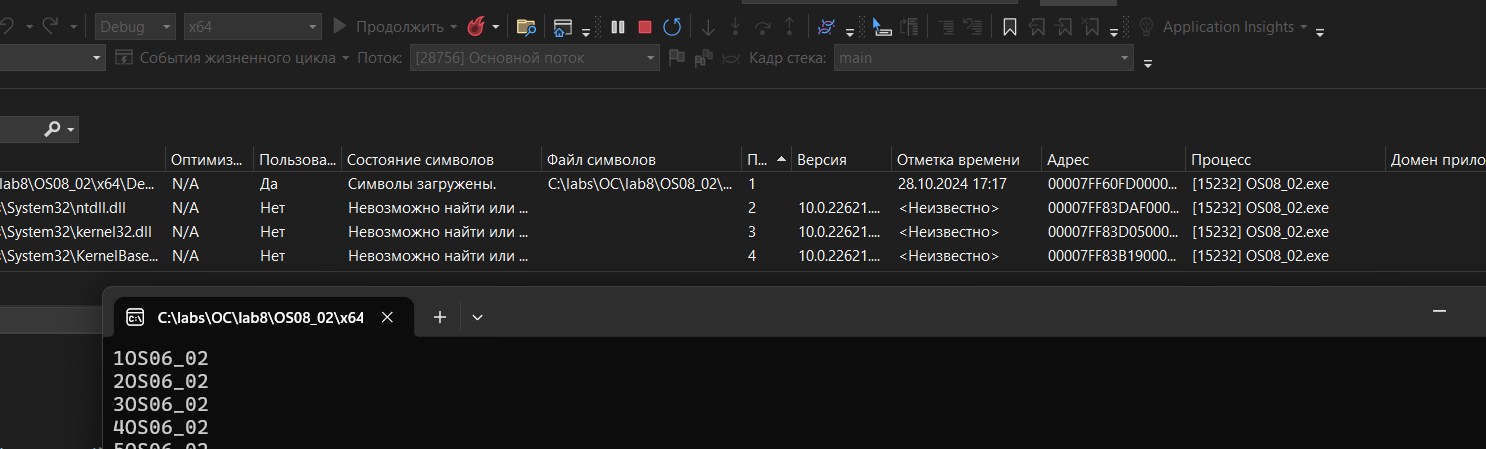


**Задание 02.Windows**

1. Разработайте консольное приложение **OS08\_02**, выполняющее длинный цикл.
2. Продемонстрируйте с помощью отладчика адреса расположения модулей приложения **OS08\_02**.
3. Установите для приложения **OS08\_02** стандартный адрес загрузки в память.
4. Продемонстрируйте с помощью отладчика стандартный адрес расположения модулей приложения **OS08\_02**







**Задание 03.Windows**

1. Разработайте консольное приложение **OS08\_03**, выполняющее получение 256 страниц оперативной памяти.
2. Разместите в этой памяти массив типа **int,** полностью занимающее все 256 страниц.
3. Заполните этот массив нарастающей последовательностью чисел с шагом 1.
4. Запишите 3 первых буквы своей фамилии в 16-ричными числами в кодировке Windows-1251.
5. Найдите в полученной области памяти с помощью отладчика значение в байте, имеющем адрес вычисленный по следующему принципу: номер страницы = число в нулевом байте, смещение в странице = число 12 бит в 1ом и втором байтах.

*Пример: Иванов*

*И = C8*

*в = E2*

*а = E0*

*Страница C8 = 200, смещение E2E = 3630*

С - 209(10) - D1(16)

е - 229(10) - E5(16)

л - 235(10) - EB(16)

Страница D1 = 209

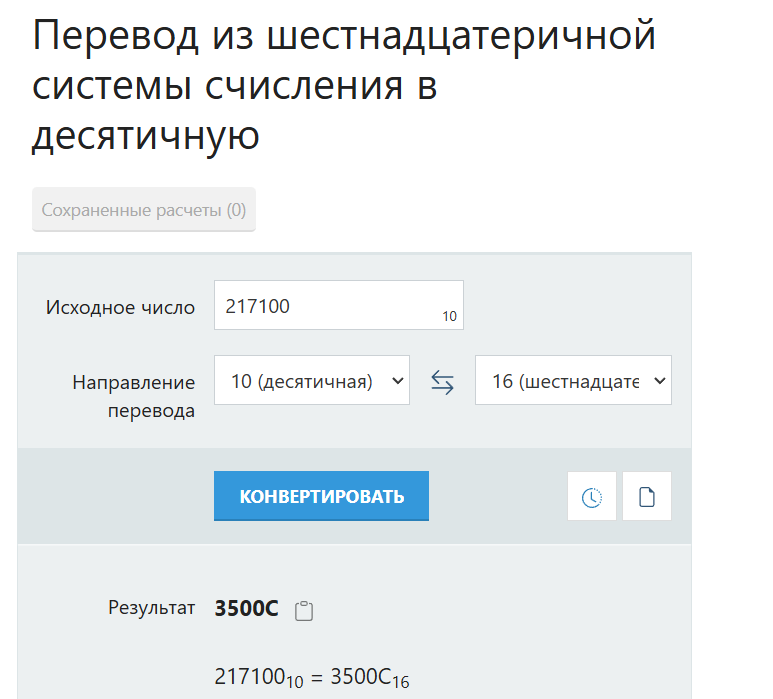
209 \* 4096 = 874240(10) = 0xD1000 - добавить для перехода на страницу

Смещение C0C = 3084(10) = 0x00000C0C

Искомое значение: начало массива + 0xD1000 + 0x00000C0C









**Задание 04.Windows**

1. Разработайте консольное приложение **OS08\_04,** которое включает функцию **sh**, принимающую 1 параметр: дескриптор (HANDLE) heap.
2. Функция **sh** выводит на консоль, общий размер heap, размеры распределенной и нераспределенных областей памяти heap.
3. Приложение **OS08\_04** размещает в стандартной heap процесса int-массив размерности 300000.
4. Выведите с помощью функции **sh** информацию до размещения массива и после.
5. Объясните результат.



**Задание 05.Windows**

1. Разработайте консольное приложение **OS08\_05** аналогичное приложению **OS06\_05,** но использующее пользовательскую heap, которая имеет первоначальный размер 4MB.
2. Объясните результат.



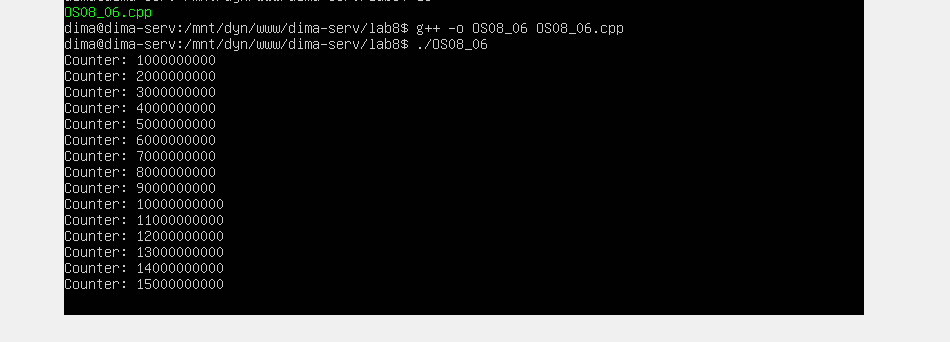
**Задание 06.Linux**

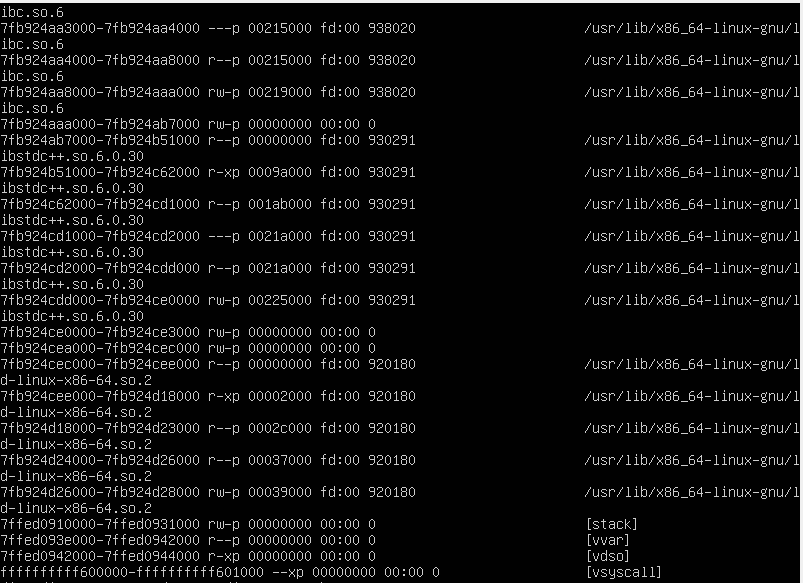
1. Разработайте консольное приложение **OS08\_06**, выполняющее длинный цикл.
2. Продемонстрируйте с помощью файловой системы **/proc** структуру адресного пространства.
3. Продемонстрируйте с помощью **pmap**  структуру адресного пространства.

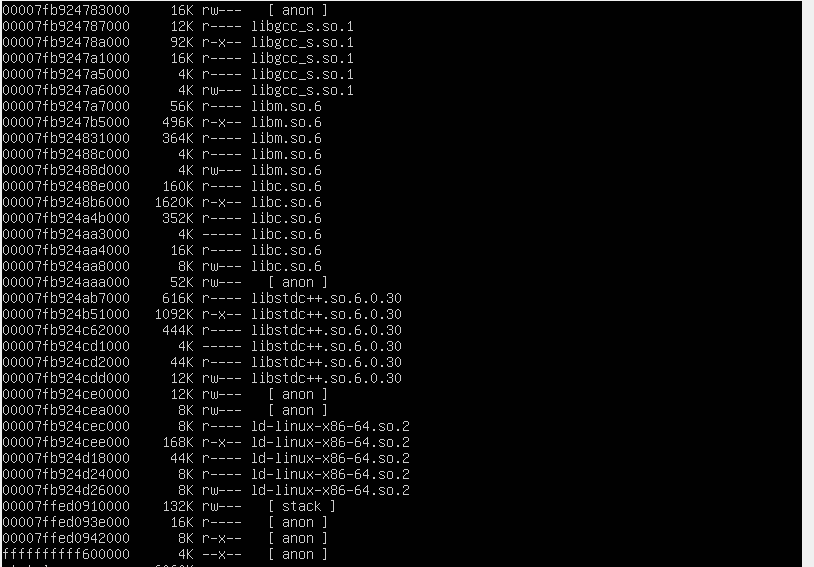
Определите с помощью утилиты objdump адрес загрузки main-модуля, секций с кодом, данными, неинициализированными глобальными переменными.













**Задание 07.Linux**

1. Разработайте консольное приложение **OS08\_07**, которое динамически выделяет 256 МБ памяти.
2. В выделенной памяти разместите int-массив максимальной размерности. Проинициализируйте массив последовательными значениями с шагом 1.
3. Выведите на консоль адрес выделенной памяти.
4. После инициализации приложение должно приостановить свое выполнение на длительный интервал времени.
5. С помощью утилиты pmap определите область памяти в которой выделена память.





