Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования "Новгородский государственный университет имени Ярослава Мудрого"

Кафедра «Информационных технологий и систем»

Дисциплина «Операционные системы»

Отчет по лабораторной работе

«Организация FS в UNIX. Работа с файлами. Понятие о memory mapped files»

Выполнил студент группы 9091

\_\_\_\_\_\_\_\_\_\_\_\_\_/Юшин Григорий Станиславович/

Подпись ФИО

Принял преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_/Ананьев Владислав Валерьевич/

Подпись ФИО

Великий Новгород

2021

**Цель лабораторной работы**

## Задание на лабораторную работу

Написать программу для работы с файлами, проецируемыми в память.

В параметрах командной строки программа должна получить имена двух файлов.

Первый файл должен содержать входные данные - короткий текст на английском языке. Достаточно нескольких предложений. (Например, их можно скопировать со страницы справки man mmap, или с любой страницы из интернета.)

Второй файл (выходной) требуется открыть, если надо, то создать, и спроецировать в память текущего процесса.

Программа должна прочитать входной файл, размещая получаемые данные в памяти, в которую спроецирован выходной файл.

В конце работы требуется освободить все выделенные ресурсы и закрыть оба файла.

**Исходный текст программ**

|  |
| --- |
| main.c |
| #include <stdio.h>  #include <stdlib.h>  #include <sys/mman.h>  #include <sys/types.h>  #include <sys/stat.h>  #include <fcntl.h>  #include <unistd.h>  int main(int argc, char\* argv[])  {  if(argc < 3) exit(-1);  const char \*inp\_file = argv[1];  const char \*out\_file = argv[2];  int fileDesc = open(out\_file, O\_RDWR | O\_CREAT, 0600);  if (fileDesc < 0)  {  perror("error with open of out file");  return -1;  }  struct stat inFileStat;  stat(inp\_file, &inFileStat);  size\_t input\_size = inFileStat.st\_size;  ftruncate(fileDesc, input\_size);  char \*mapPtr = mmap(0,input\_size,PROT\_WRITE,MAP\_SHARED,fileDesc,0);  close(fileDesc);  fileDesc = open(inp\_file, O\_RDONLY);  read(fileDesc, mapPtr, input\_size);  close(fileDesc);  if(munmap((void \*)mapPtr, input\_size) != 0)  {  perror("error with writing in out file");  return -1;  }  } |

**Логика приложения:**

1. Считываем входные параметры
2. Открываем (создаём) выходной файл на запись.
3. Узнаём размер входного файла.
4. Создаём проецируемую память в пространстве процесса.
5. Закрываем выходной файл.
6. Открываем входной файл.
7. Считываем вызовом read() весь файл в проецируемую память.
8. Закрываем входной файл.
9. Unmap
10. ???
11. PROFIT!

**Результат выполнения программ**

|  |
| --- |
| Файл input.txt |
| The form, design, and implementation of CPUs have changed over time, but their fundamental operation remains almost unchanged. Principal components of a CPU include the arithmetic logic unit (ALU) that performs arithmetic and logic operations, processor registers that supply operands to the ALU and store the results of ALU operations, and a control unit that orchestrates the fetching (from memory) and execution of instructions by directing the coordinated operations of the ALU, registers and other components. |
| Файл output.txt |
| The form, design, and implementation of CPUs have changed over time, but their fundamental operation remains almost unchanged. Principal components of a CPU include the arithmetic logic unit (ALU) that performs arithmetic and logic operations, processor registers that supply operands to the ALU and store the results of ALU operations, and a control unit that orchestrates the fetching (from memory) and execution of instructions by directing the coordinated operations of the ALU, registers and other components. |

**Вывод**

