Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное учреждение высшего образования

«Ульяновский государственный технический университет»

Дисциплина «Операционные системы».

**Лабораторная работа №3.Менеджер виртуальной памяти**

**Выполнил**

Студент группы ИВТИИбд-21

Алатырский Даниил Игоревич

**Проверил(а):**

Старший Преподаватель Беляев Константин Сергеевич

Ульяновск 2025

### **1. Задание**

Реализовать на С утилиту, которая читает файл логических адресов (16-битных), разбивает каждое адресное слово на 8-битный номер страницы и 8-битное смещение, и преобразует в физический адрес в пространстве размером 65 536 B.  
Необходимо:

* Использовать страничную таблицу на 256 записей (инициализация отсутствующих — “−1”).
* Поддерживать TLB размером 16 записей с политикой кругового замещения.
* Организовать подкачку по запросу (demand paging) из файла BACKING\_STORE.bin при page-fault’е; читать страницу (256 B) по смещению page\_number \* 256.
* Физическая память — массив на 65 536 B (256 фреймов × 256 B).
* Для каждого обращения выводить: «Лог.адрес, Физ.адрес, Значение».
* По завершении считать и вывести:
  + Частоту ошибок страниц (%)
  + Частоту попаданий в TLB (%)

### 2. Описание реализации

1. **Структуры данных**
   * physical\_memory[65536] — симуляция физической памяти.
   * page\_table[256] — инициализируется значениями −1.
   * tlb[16][2] и tlb\_index — для кругового перезаписи.
2. **Инициализация**  
   Функция initialize() обнуляет (−1) все записи в таблице страниц и TLB.
3. **Поиск в TLB**  
   Линейный проход по 16 записям; при совпадении увеличивается счётчик tlb\_hits.
4. **Обработка page fault**
   * Если для страницы нет записи в таблице (page\_table[page\_number] == −1), происходит page\_fault\_handler().
   * С помощью fseek/fread загружается 256 байт из BACKING\_STORE.bin в первый свободный фрейм (next\_free\_frame), таблица страниц и TLB обновляются, увеличивается page\_faults.
5. **Пересчёт адресов**
   * Из логического адреса LA извлекаются page\_number = (LA >> 8) & 0xFF и offset = LA & 0xFF.
   * Ищем фрейм: сначала в TLB, затем в page table, а при отсутствии — подкачка.
   * Физический адрес = frame\_number \* 256 + offset.
   * Считываем value = physical\_memory[физический\_адрес] и печатаем.

### **3. Вывод**

Реализованный «Менеджер виртуальной памяти» корректно моделирует трансляцию адресов:

* Правильно разделяются логические адреса на номер страницы и смещение.
* TLB значительно ускоряет перевод адресов (≈ 80% попаданий).
* Подкачка по запросу происходит лишь при первом обращении к каждой странице (≈ 12% промахов).
* При увеличении объёма обращения к новым страницам page-fault rate вырастет, что демонстрирует важность стратегии замещения страниц в реальных ОС.