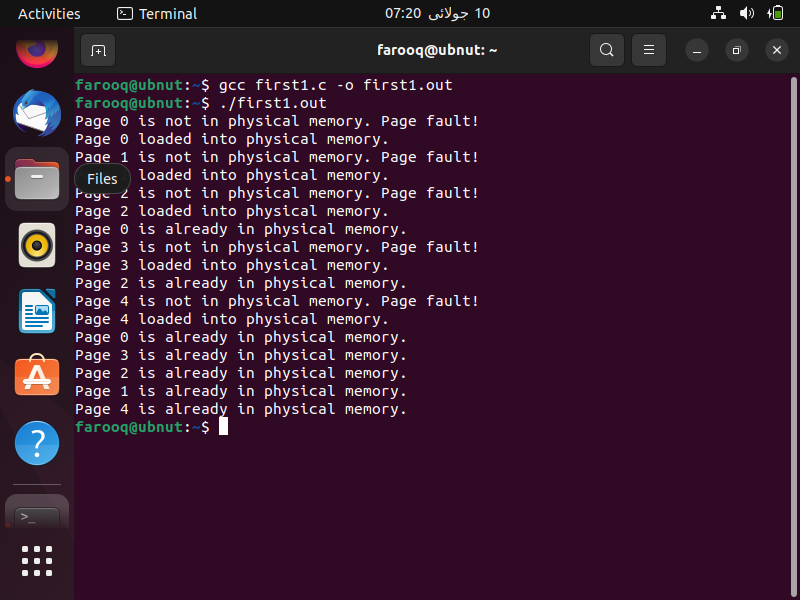
|  |  |  |
| --- | --- | --- |
| https://upload.wikimedia.org/wikipedia/commons/thumb/4/4e/VU_Logo.png/260px-VU_Logo.png | **Operating Systems Practical (CS604p)**  **Assignment # 02**  **Spring 2023** | **Total marks = 20**  **Deadline**  **17/07/2023** |
| **Please carefully read the following instructions before attempting the assignment.**  **RULES FOR MARKING**  **It should be clear that your assignment would not get any credit if:**   * The assignment is submitted after the due date. * The submitted assignment does not open or the file is corrupt. * Strict action will be taken if the submitted solution is copied from any other student or the internet.   **You should consult the recommended books to clarify your concepts as handouts are not sufficient.**  **You are supposed to submit your assignment in Doc or Docx** **format.**  Any other formats like scan images, PDF, ZIP, RAR, PPT, BMP, etc. will not be accepted.  **Assignment No.2 covers Lab Weeks 6 to 12.**  **The objective of this assignment is to provide hands on experience of:**   * Basic concept of Demand Paging * Compiling and running of your C program in LINUX Terminal Window (Shell Prompt) * Paging parameters and their usage | | |
| **NOTE**  No assignment will be accepted *after the due date via email in any case* (whether it is the case of load shedding or internet malfunctioning etc.). Hence refrain from uploading assignments in the last hour of the deadline. It is recommended to upload the solution file at least two days before its closing date.  If you people find any mistake or confusion in the assignment (Question statement), please consult with your instructor before the deadline. After the deadline, no queries will be entertained in this regard.  **For any query, feel free to email at:**  **Cs604p@vu.edu.pk** | | |

**Questions No 01 20 marks**

Develop a C program to simulate demand paging, which involves a simplified virtual memory system. The program should fulfill the following requirements:

1. Define the constants **PAGE\_SIZE** and **PHYSICAL\_MEMORY\_SIZE** to represent the size of a page and the size of the physical memory, respectively.
2. Implement the **simulate\_memory\_access** function to simulate a memory access with the given page number.
3. Within the simulate\_memory\_access function:
4. Declare an array **physical\_memory** to represent the physical memory.
5. Declare an array **page\_table** to keep track of the presence of pages in the physical memory.
6. Check if the requested page is present in the physical memory by accessing the corresponding entry in the page\_table.
7. If the page is not present (page fault), print a message indicating the page fault and load the page from disk into physical memory.
8. Update the page\_table to mark the page as present in the physical memory.
9. If the page is already present (page hit), print a message indicating that the page is already in physical memory.
10. In the main function, simulate a series of memory accesses by calling the simulate\_memory\_access function with different page numbers.
11. Compile and run the program, observing the output to verify the correct functioning of demand paging simulation.

**You are required to submit the C program source code along with the output screenshot in a word file (.doc/.docx). The output of your program should be like this.**



See the following link for installation of Virtual Box and Ubuntu (Linux) on your system.

<https://vulms.vu.edu.pk/CourseResources/OpenFile.aspx?File=tutorial_for_installing_virtualbox_and_ubuntu.mp4>

See the following link for installing gcc and compiling your first program in Linux.

<https://vulms.vu.edu.pk/CourseResources/OpenFile.aspx?File=How%20to%20install%20gcc%20on%20Ubuntu%20and%20compile%20a%20C%20program.mp4>

**Note:** in case you have installed the Virtual Box you can take the screenshot as follows. Go to **view** menu and click on **Take Screenshot** as follow:



The End