

Assignment for 3rd Year 1st Semester Students

IT/PC/B/S/313

GUIDELINES

Try to write a clean program with enough comments.

At the beginning of the file, use block comments to write details about name, roll no, assignment details, input required and output generated.(as per Comment_File)

Also put the compilation [should be WARNING free] and execution sequence under the block comment. .(as per Comment_File)

The name of the file should be as per the following format.

<Two digit Team No>_<Assignment No>

The type of the file should be .c

The assignment files should be uploaded using google form which I will email you later.

While coding, always use indentation of 4 spaces.

Blocks of code should be separated by a newline.

Always use command line argument handling to take inputs.

Late Submission will incur penalties.

Duplicate assignments will incur penalties.
[Marks will be allocated proportionately]

Not adhering to any of these guidelines will incur penalties.

For the description of any system/library call use man command.

Always use 'perror' routine to check the return status of the system/library call.

ASSIGNMENT – 4 **(mmap and page fault)**

Total Marks - 10 [CO2]

Objective of this programming assignment is to use `mmap()` call and observe page-fault using the 'sar' command.

A big file (about 8GB) should be created using the 'fallocate' command. This big file should be written with a single byte value (say X) at a specific offset (say F). Both the values and the offset should be generated using a random function. Please do remember this random function should generate a quantity anywhere between 0 and 8G for the value of F and between 0-255 for the value of X.

The above big file should also be mapped in the virtual address space using `mmap()` call. Once it is mapped, the data should be read from the same specific offset (F). Now, if the data read is X'; then verify that X and X' are the same. In case of verification failure, an error message is to be printed and also the program should terminate. Note that, the offset value F can be anywhere between 0 and 8G and you should display the offset as hex number format.

This sequence of writing and reading data to/from a specific offset and also verification should be put in a while loop to go forever.

In another terminal execute the command 'sar -B 1' to observe for the page fault. This command should be started before the above program is put under execution. So, one can observe that the page faults are increasing, once the above program starts executing.

The output of the program and the 'sar' command should be pasted as a comment at the beginning of the program file as indicated by the guidelines.