**Session 2025-2026**

|  |  |
| --- | --- |
| **Vision:** Dream of where you want. | **Mission:** Means to achieve Vision |

**Program Educational Objectives of the program (PEO):** (broad statements that describe the professional and career accomplishments)

|  |  |  |  |
| --- | --- | --- | --- |
| PEO1 | **Preparation** | **P: Preparation** | **Pep-CL abbreviation**  **pronounce as Pepsil easy to recall** |
| PEO2 | **Core Competence** | **E: Environment (Learning Environment)** |
| PEO3 | **Breadth** | **P: Professionalism** |
| PEO4 | **Professionalism** | **C: Core Competence** |
| PEO5 | **Learning Environment** | **L: Breadth (Learning in diverse areas)** |

**Program Outcomes (PO):** (statements that describe what a student should be able to do and know by the end of a program)

**Keywords of POs:**

Engineering knowledge, Problem analysis, Design/development of solutions, Conduct Investigations of Complex Problems, Engineering Tool Usage, The Engineer and The World, Ethics, Individual and Collaborative Team work, Communication, Project Management and Finance, Life-Long Learning

**PSO Keywords:** Cutting edge technologies, Research

“I am an engineer, and I know how to apply engineering knowledge to investigate, analyse and design solutions to complex problems using tools for entire world following all ethics in a collaborative way with proper management skills throughout my life.” *to contribute to the development of cutting-edge technologies and Research*.

**Integrity:** I will adhere to the Laboratory Code of Conduct and ethics in its entirety.

**Name and Signature of Student and Date**

(Signature and Date in Handwritten)

Aryan Sukhdewe

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Session** | **2025-26 (ODD)** | | **Course Name** | **HPC Lab** | |
| **Semester** | **7** | | **Course Code** | **22ADS706** | |
| **Roll No** | **40** | | **Name of Student** | **Aryan Sukhdewe** | |
|  |  | |  |  |  |
| Practical Number | | **1** | | | |
| Course Outcome | | 1. Understand and Apply Parallel Programming Concepts  2. Analyze and Improve Program Performance.  3. Demonstrate Practical Skills in HPC Tools and Environments. | | | |
| Aim | | Introduction to Linux and HPC Environment | | | |
| Problem Definition | | To Solve large-scale scientific, engineering, and  data-intensive problems  To Reduce processing time  To Improve simulation accuracy | | | |
| Theory  (100 words) | | **Definition:**  High Performance Computing (HPC) refers to the use of  supercomputers and parallel processing techniques to  solve complex computational problems faster and more  efficiently than traditional systems.  **Key Components of HPC Systems**  Compute Nodes (Processors/CPUs/GPUs)  Memory (RAM)  Storage (Disks/SSDs)  Interconnect (High-speed Network)  Software Stack (Compilers, Libraries, Tools)  High Performance Computing (HPC) leverages supercomputers and parallel processing to solve complex, large-scale problems efficiently. HPC systems consist of compute nodes, memory, storage, high-speed interconnects, and software stacks. Parallel computing, including data and task parallelism, is central to HPC, using shared memory (OpenMP) or distributed memory (MPI) models, and GPU programming (CUDA/OpenCL). Linux is the preferred OS for HPC due to its open-source nature and robust support. Essential Linux commands help manage files and processes. Job schedulers like SLURM efficiently manage HPC workloads by submitting, monitoring, and controlling batch jobs via scripts. | | | |
| Procedure and Execution  (100 Words) | | Algorithm:  Step 1: Prepare your HPC job script (.h file) specifying resources and commands.  Step 2: Use Linux commands to navigate and manage files.  Step 3: Submit the job script using SLURM’s batch command.  Step 4: Monitor the job status using queue or run for interactive jobs.  Step 5: If needed, cancel jobs using cancel command.  Step 6: Analyze job output once execution completes. | | | |
| Code:  #!/bin/bash  #SBATCH --job-name=example\_job # Job name  #SBATCH --output=output\_%j.txt # Output file (%j = job ID)  #SBATCH --ntasks=4 # Number of tasks (processes)  #SBATCH --time=00:10:00 # Time limit (HH:MM:SS)  #SBATCH --partition=compute # Partition or queue name  # Load required modules (if any)  module load gcc  # Run your parallel program (example using MPI)  mpirun ./my\_parallel\_program | | | |
| Output:  **On Next page** | | | |
| Output Analysis | | Upon submission, SLURM assigns a job ID and queues the job.  Use queue to check job status (pending, running, completed).  After job completion, check output file output\_<jobID>.txt.  Analyze results or logs for correctness and performance.  If job fails, review error messages and modify the script or program accordingly. | | | |
| Link of student Github profile where lab assignment has been uploaded | | **https://github.com/aryanycce/HPC-Practicals** | | | |
| Conclusion | | HPC leverages parallel computing on Linux clusters to efficiently solve complex scientific and engineering problems. Mastery of Linux commands, parallel programming models, and job scheduling systems like SLURM is essential. This environment enables better performance, scalability, and resource management, making HPC indispensable in data-intensive tasks and simulations. | | | |
| Plag Report (Similarity index < 12%) | |  | | | |
| Date | | **26/08/25** | | | |