CDSC 604: Parallel Programming (7 ECTS)

I. Aims

This module aims to enable students to acquire a working knowledge of Parallel programming, which they can apply to solve a variety of scientific problems.

II. Contents

This course is intended for students who are interested in learning how to take advantage of parallel and distributed computing with the focus of writing parallel code for processor-intensive applications to be run on clusters, the grid, the cloud, or shared infrastructure. The objectives of this course are to give the students an understanding of how they can use parallel computing in their research and enable them to write parallel code for their high-performance computing applications in Unix and Linux operating systems environments. Extensive use of pertinent and practical examples from scientific computing will be made throughout. The programming languages used will be Fortran, or C. Both the shared and distributed paradigms of parallel computing will be covered via the OpenMP and MPI libraries.

III. Learning Outcomes

Subject-specific Knowledge:

- After studying these module students will be:

  Comfortable to use OpenMP and MPI

  Familiar with some of the fundamentals of Parallel programming which is useful for solving various physical problems.

Subject-specific Skills:

- In addition to the acquisition of subject-specific knowledge, students will be able to apply decomposing tasks, understanding task dependency, and data dependency.

IV. Modes of Teaching and Learning

- The mode of the teaching-learning process will involve lectures (50%) and practical work (50%) on a computer or cluster which involves about 80% student participation.
- The lecture will be offered through teaching, one and a half hours per day, and two days a week for a semester.

- Two weeks are reserved for examinations and processing of the examinations. The lectures provide the means to give a concise, focused presentation of the subject matter module. When appropriate, the lectures will be supported by the distribution of written materials, or by relevant links on websites.
- Regular problem exercises will give students the chance to develop their theoretical understanding and problem-solving skills.
- Students will be able to obtain further help in their studies by approaching their lectures, either after lectures or at other mutually convenient times.

References

- Peter S. Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2011.
- Parallel Programming – Techniques and Applications Using Networked Workstations and Parallel Computers, Barry Wilkinson and Michael Allen, Prentice-Hall, 1999.
- Multi-Core Programming - Increasing Performance through Software Multi-Threading, Shameem Akhter and Jason Roberts, Intel Press 2006.
- Parallel Programming in C with MPI and OpenMP, Michael J. Quinn, McGraw Hill 2003.