# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3730209**
**Semester III**
**SUBJECT NAME: High Performance Scientific Computing**

**Type of course: Elective**

**Prerequisite:** Linear Algebra and Numerical Methods, Parallel Algorithms

**Rationale:** Computation and simulation are increasingly important in all aspects of science and engineering. At the same time writing efficient computer programs to take full advantage of current computers is becoming increasingly difficult. Even laptops now have 4 or more processors, but using them all to solve a single problem faster often requires rethinking the algorithm to introduce parallelism, and then programming in a language that can express this parallelism. Writing efficient programs also requires some knowledge of machine arithmetic, computer architecture, and memory hierarchies. High performance *programming* is an important aspect of high performance scientific computing, and so the main theme of the course is the use of basic tools and techniques to improve your efficiency as a computational scientist.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 0 | 3 | 70 | 30 | 0 | 0 | 100 |

**Content:**

| Sr. No. | Content | Total Hrs | % Weightage |
|---|---|---|---|
| 1 | Introduction: Single Processor Computing, Parallel Computing, Parallel System Organization | 3 | 6 |
| 2 | Numerical Linear Algebra, High Performance Linear Algebra, Numerical Treatment of Differential Equations | 5 | 10 |
| 3 | Applications: Molecular Dynamics, Sorting, Graph Analytics, N-body Problems, Monte Carlo Methods, Computation Biology | 7 | 15 |
| 4 | Interactive Python using IPython, and the IPython Notebook, Python scripting and its uses in scientific computing, Subtleties of computer arithmetic that can affect program correctness | 10 | 21 |
| 5 | Fortran 90, a compiled language that is widely used in scientific computing, Makefiles for building software and checking dependencies, Analyse the cost of data communication. Registers, cache, main memory, and how this memory hierarchy affects code performance. | 10 | 21 |
| **6** | OpenMP on top of Fortran for parallel programming of shared memory computers, such as a multicore laptop., MPI on top of Fortran for distributed memory parallel programming, such as on a cluster, Parallel computing in IPython, Debuggers, unit tests, regression tests, verification and validation of computer codes, Graphics and visualization of computational results using Python | 13 | 27 |
| | **Total** | 48 | **100%** |

After learning the course the students should be able to:

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|

| | | |
|---|---|---|
| CO-1 | demonstrate the knowledge of numerical computing using an appropriate programming language. | 10% |
| CO-2 | be competent in experimental computing in a numerical context and of the optimization of algorithms on high performance architectures. | 20% |
| CO-3 | be able to reason about the accuracy of mathematical and numerical models of real physical phenomena. | 25% |
| CO-4 | have an awareness of the modern field of computational science and engineering and of the impact of high performance computing on science and industry. | 25% |
| CO-5 | have an understanding of the various paradigms of high performance computing and their potential for performance and programmability. | 20% |

Distribution of marks weightage for cognitive level

| **Bloom's Taxonomy for Cognitive Domain** | **Marks % weightage** |
|---|---|
| Recall | 5 |
| Comprehension | 10 |
| Application | 20 |
| Analysis | 25 |
| Evaluate | 25 |
| Create | 15 |

**Reference Books:**


1. Scott, Clark, and Bagheri, *Scientific Parallel Computing*.
2. Parallel Programming for Multicore and Cluster Systems by Thomas Rauber and Gudula Runger.
3. Using OpenMP: Portable Shared Memory Parallel Programming by Chapman, Jost, and vander Pas.
4. An Introduction to High Performance Scientific Computing, Lloyds D. Fosdick, Elizabeth R. Jessup, Carolyn J. C. Schauble, Gitta Domik
5. Heath, M.T. Scientific Computation - An Introductory Survey, McGraw-Hill, 1997.
6. Buyya, R. High Performance Cluster Computing: Programming and Applications, Prentice Hall, Upper Saddle River, New Jersey 1999.s