Title of the Course: High Performance Parallel Programming

1. Credit requirement: (L-T-P: 3-1-0, Credit: 4)
2. Please select the committee for Approval: PGPEC
3. Name of the Dept: CSE
4. Please Specify the Level of the Subject: PG level
5. Semester: SPRING
6. Whether the subject will be offered as compulsory or elective: Elective
7. Prerequisite(s) for the subject, if any (Please give the subject numbers and names): Basic knowledge in programming and computer systems (fitness to be decided by course instructors)
8. **Course Objective**

   Writing efficient parallel programs is a requirement in a wide array of domains including scientific and engineering computation, distributed machine learning applications and computational science. There exists a wide choice of programming models, development environments, runtime systems, optimizations and high performance architectural targets. The objective of this course is to familiarize students with the above topics so that they become conversant with efficient implementation of large scale parallel programs.

9. **Study Materials**
   (a) "Computer Architecture -- A Quantitative Approach" - John L. Hennessy and David A. Patterson
   (b) "Heterogeneous Computing with OpenCL" -- Benedict Gaster, Lee Howes, David R. Kaeli
   (c) CUDA Reference manual
   (d) "Hadoop: The Definitive Guide, 4th Edition" – Tom White.
   (e) Spark Programming Guide.
   (f) "Using OpenMP" by Barbara Chapman, Gabriele Jost and Ruud van der Pas
   (g) "MPI: The Complete Reference" by Marc Snir, Jack Dongarra, Janusz S. Kowalik, Steven Huss-Lederman, Steve W. Otto, David W. Walker
   (h) "Parallel Programming with MPI" by Peter Pacheco
   (i) Web resource: http://openmp.org/wp/
10. **Syllabus (with Lecture wise break-ups)**

(a) **Introduction to HPC Systems[12]**

i. Introduction to basic architecture and OS concepts [2]
ii. Multi-core CPUs [2]
iii. High-speed interconnects [2]
iv. High performance file systems [2]
v. GPU systems [2]
vi. High performance clusters [2]

(b) **Parallel Programming Models, Runtime Systems [18]**

i. OpenMP and MPI [4], Thread Management, Workload manager and Job Schedulers [2]
ii. CUDA / OpenCL [4], HW schedulers, Software runtime systems [2]
iii. MapReduce System Architecture and runtime management [6]

(c) **Benchmarks and Case Studies [6]**

i. Developer Libraries: [2]
   A. Linear Algebra : MKL, BLAS, Lapack
   B. Distributed Machine Learning using Spark Framework
ii. Benchmarks: [2]
   A. GPU Benchmarks : Parboil, Rhodinia
   B. Computational Science : BLAST and SOM (Bioinformatics), Fluid Dynamics (CFD packages)
iii. Hybrid Parallel Programming: Putting it together (Python, MPI, OpenMP, CUDA) [2]

(d) **Tutorials [12]**

i. Familiarization with HPC softwares : OpenMP and MPI, Spark Framework for Map-Reduce [2]
ii. Benchmark based performance evaluation experiments on HPC systems [2]
iii. HPC Application development: Drug design, Fault Simulation, Machine Learning Application development [8]

11. **Names of the faculty members of the Department/Centers/School who have the necessary expertise and will be the willing to teach the subject (Minimum two faculty members should be willing to teach the subject)**

(a) Pralay Mitra
(b) Sourangshu Bhattacharya
(c) Soumyajit Dey
(d) Arobinda Gupta

12. **Do the contents of the subject have an overlap with any other subject offered in the Institute?**

    (a) **Approximate percentage of overlap:** No significant overlap with existing subjects.

    (b) **Reasons for offering the new subject in spite of the overlap : NA**