1.	Title of the Course	GPU Computing
2.	Course Number	CS5221
3.	Status of the Course	Elective
4.	Structure of Credits	3-0-0-3
5.	Offered To	PG
6.	New Course/Modification to	New
7.	To be Offered by	Department of Computer Science and Engineering
8.	To take effect from	January 2020
9.	Prerequisite	CoT for UG
10.	Whether approved by the Department	Yes
11.	**Course Objective:** To design parallel algorithms, implement them on graphics processing units (GPUs), and improve their performance by utilizing the GPU architecture effectively.	
12.	**Course Content:** Introduction: General purpose programming in graphics processing units (GPU), GPU architecture, compute unified device architecture (CUDA); Data parallelism: data transfer, kernel functions, CUDA thread organization, thread hierarchy, warps, blocks, grids, streaming multiprocessors; Memory: memory hierarchy, global, shared, local, textures, constant memory, bank conflicts; Synchronization: memory consistency, barriers (local versus global), atomics, memory fence, worklists, synchronization across Central Processing Unit (CPU) and GPU, device functions; Parallel patterns: convolution, prefix sum, reduction; Advanced concepts: performance tuning, asynchronous processing, task-dependence, streams, overlapping data transfer and kernel execution, graph algorithms using GPU processing, dynamic parallelism, unified virtual memory, multi-GPU processing.	
13.	**Text book(s):** 1. Kirk D and Hwu W, *Programming Massively Parallel Processors: A Hands-on Approach*, Morgan Kaufman, 2nd Edition (2015).	
14.	**Reference(s):** 1. Cheng J, Grossman M and McKercher T, *Professional CUDA C Programming*, Wrox Press Ltd. (2014). 2. Cook S, *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs*, Morgan Kaufman (2012).	