Department of Computing Technologies

Honors in GPU-Accelerated Computing

Curriculum for Honors in GPU Accelerated Computing					
Course Code	Course Title	Hours/ Week			C
		L	T	P	
Preparatory Course					
&21HCSP001	*Basics of Computer Architecture and Organization*	*2*	*0*	*0*	*2*
Foundation Courses					
21HCSF001	*Advanced High Performance Computing*	*3*	*0*	*0*	*3*
21HCSF002	*Introduction to Parallel Programming with OpenMP and MPI Using C and Python*	*3*	*0*	*2*	*4*
21HCSF003	*Fundamentals of Accelerated Computing with CUDA Using Python*	*2*	*0*	*2*	*3*
Elective Courses (Any three for 8 credits)					
21HCSE001	*Advanced Computer Architecture*	*3*	*0*	*0*	*3*
21HCSE002	*GPU Computing*	*2*	*0*	*2*	*3*
21HCSE003	*Accelerated Deep Learning*	*2*	*0*	*2*	*3*
21HCSE004	*Introduction to Concurrent Programming with GPUs*	*2*	*0*	*2*	*3*
21HCSE005	*Applied Accelerated Artificial Intelligence*	*2*	*0*	*2*	*3*
$21HCSE006	*Fundamentals of Accelerated Computing with OpenACC*	*2*	*0*	*2*	*3*
21HCSE007	*Sparse Intensive GPU Computing*	*2*	*0*	*0*	*2*
21HCSE008	*NVIDIA GPU's Accelerated Computing*	*2*	*0*	*1*	*2*
$21HCSE009	*Introduction to Python Programming*	*2*	*0*	*1*	*2*
Total Learning Credits					**20**

Course Code	21HCSP001	Course Name	Basics of Computer Architecture and Organization	Course Category	P	Preparatory Course	L	T	P	C
							2	0	0	2

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil

Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil

Course Learning Rationale :	The purpose of learning this course is to:
CLR-1 :	To impart basic concepts of computer architecture and organization
CLR-2 :	To explain key skills of constructing cost-effective computer systems
CLR-3 :	To familiarize the basic CPU organization.
CLR-4 :	To help students in understanding various memory devices.
CLR-5 :	To help students in understanding various memory devices.

Course Outcomes (CO):	At the end of this course, learners will be able to:
CO-1 :	Identify various components of computer and their interconnection
CO-2 :	Identify basic components and design of the CPU: the ALU and control unit.
CO-3 :	Compare and select various Memory devices as per requirement.
CO-4 :	Compare various types of IO mapping techniques
CO-5 :	Critique the performance issues of cache memory and virtual memory

Program Outcomes (PO)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
3	2	-	-	-	-	-	-	-	-	-	-	-	-	-
3	2	-	-	-	-	-	-	-	-	-	-	-	-	-
3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
3	1	2	-	-	-	-	-	-	-	-	-	-	-	-
3	2	1	-	-	-	-	-	-	-	-	-	-	-	-

Unit-1 ***Structure of Computers***
Computer types, Functional units, Basic operational concepts, VonNeumann Architecture, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Data representation, Fixed and Floating point, Error detection and correction codes.

Unit-2 ***Basic Computer Organization and Design***
Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs RISC

Unit-3 ***Registers and Micro-Operations***
Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit.

Unit-4 ***Memory System***
Memory Hierarchy, Semiconductor Memories, RAM(Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID.

Unit-5 ***Input Output and Multiprocessors***
I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. MULTIPROCESSORS: Characteristics of multiprocessors, Interconnection structures, Inter Processor Arbitration, Inter processor Communication and Synchronization, Cache Coherence.

Learning Resources

1. Georg Hager Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, Chapman & Hall/CRC Computational Science Series,2018
2. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012
3. https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html
4. Nicholas Wilt, ―CUDA Handbook: A Comprehensive Guide to GPU Programming, Addison – Wesley, 2013.
5. JASON SANDERS EDWARD KANDROT ,CUDA by Example A general purpose , NVIDIA

Bloom's Level of Thinking		Continuous Learning Assessment (60% weightage)		Final Examination (40% weightage)
		CLA – 1 (50%)	CLA – 2 (10%)	
		Theory	Theory	Theory
Level 1	Remember	*20%*	*20%*	*20%*
Level 2	Understand	*20%*	*20%*	*20%*
Level 3	Apply	*50%*	*50%*	*50%*
Level 4	Analyze	10%	10%	10%
Level 5	Evaluate			
Level 6	Create			
Total		100 %	100 %	100 %

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Mr.Vimal,CDAC	1. Dr.Noor Mohammed, IIITDM,Kancheepuram	1. Dr.S.Nagadevi, Assistant Professor,SRMIST,Kattankulathur
2. Mrs.V.A.Prabha,CDAC	3. Dr.Mohammed Hazan, IIITDM,Kancheepuram	4. Dr.M.Revathi, Assistant Professor,SRMIST,Kattankulathur

Course Code	21HCSF001	Course Name	Advanced High Performance Computing	Course Category	F	Foundation Course	L	T	P	C
							3	0	0	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil

Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil

Course Learning Rationale :	The purpose of learning this course is to:
CLR-1 :	To understand High Performance Computing (HPC) system architectures and various computational models.
CLR-2 :	Illustrate the parallel execution models and methodologies for parallel programming and parallel applications development.
CLR-3 :	To design and implement compute intensive applications on HPC platform.
CLR-4 :	To Identify the basics of CUDA programming.
CLR-5 :	To Evaluate the GPU Performance

Course Outcomes (CO):	At the end of this course, learners will be able to:	Program Outcomes (PO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design,	Modern Tool Usage	Society & Culture	Environment &	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CO-1 :	Exhibit knowledge of Modern processors and concepts	3	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CO-2 :	Acquire the ability to process data using parallelization	3	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CO-3 :	Exhibit the knowledge on GPU architectures	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CO-4 :	Apply the knowledge on parallel programming using CUDA	3	1	2	-	-	-	-	-	-	-	-	-	-	-	-
CO-5 :	Able to study the performance of GPU	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-

Unit-1
Stored Program Computer Architecture,General-ptuposecachebased microprocessor architecture, Performance based metrics and Benchmarks, Transistors galore:Moore's Law,Pipelining, Superscalarity,SIMD,Memory hierarchies,Cache,Cache mapping,Prefetch,Multicore processors,Multithreaded processors,Vector processors

Unit-2
Taxonomy of parallel computing paradigms,Shared-memory computers,Cache coherence,UMA – ccNUMA,Distributed-memory computers,Networks- Basic performance characteristics of networks, Buses, Switched and fat-tree networks,Mesh networks, Hybrids,Parallelism- Data parallelism,Functional parallelism,Parallel scalability,Load imbalance, Design issues in parallel computing

Unit-3
Trends in CPU and GPU performance – serial versus parallel programming and Flynn's Taxonomy-- -- History of GPUs Evolution of GPU architectures – Understanding Parallelism with GPU –Typical GPU Architecture – Heterogeneous Programming-Multi-GPU systems-Multiple node systems

Unit-4
CUDA Hardware Overview – Threads, Blocks, Grids, Warps, Scheduling – CUDA Memory Model – Built-in variables and functions – thread scheduling – thread synchronization --- vector addition -- matrix multiplication

Unit-5
Parallel reduction – Memory Coalescing – loop unrolling – thread granularity --Parallel Patterns: Convolution, Prefix Sum, Sparse Matrix – Matrix Multiplication – Programming Heterogeneous Cluster

Learning Resources

1. Georg Hager Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, Chapman & Hall/CRC Computational Science Series,2018
2. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012
3. https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html
4. Nicholas Wilt, ―CUDA Handbook: A Comprehensive Guide to GPU Programming, Addison – Wesley, 2013.
5. JASON SANDERS EDWARD KANDROT ,CUDA by Example A general purpose , NVIDIA

Bloom's Level of Thinking		Continuous Learning Assessment (60% weightage)		Final Examination (40% weightage)
		CLA – 1 (50%)	CLA – 2 (10%)	
		Theory	Theory	Theory
Level 1	Remember	20%	20%	20%
Level 2	Understand	20%	20%	20%
Level 3	Apply	50%	50%	50%
Level 4	Analyze	10%	10%	10%
Level 5	Evaluate	-	-	-
Level 6	Create	-	-	-
Total		100 %	100 %	100 %

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.Vimal,CDAC	Dr.Noor Mohammed, IIITDM,Kancheepuram	Dr.S.Nagadevi, Assistant Professor,SRMIST,Kattankulathur
Mrs.V.A.Prabha,CDAC	Dr.Mohammed Hazan, IIITDM,Kancheepuram	Dr.M.Revathi, Assistant Professor,SRMIST,Kattankulathur

Course Code	21HCSF002	Course Name	Introduction to Parallel Programming with OpenMP and MPI Using C and Python	Course Category	F	Foundation Course	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	To understand the basics of parallel programming and OpenMP programming
CLR-2 :	To gain knowledge on parallel loops and schedules
CLR-3 :	To understand the basic building blocks of MPI
CLR-4 :	To understand MPI blocking and non-blocking calls to send and receive data
CLR-5 :	To evaluate the performance of OpenMP and MPI on data structures like BFS, MST

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:
CLO-1 :	Understand the basics of OpenMP parallel programming model
CLO-2 :	Acquire the knowledge of parallel loops and schedules
CLO-3 :	Illustrate the basic building blocks of MPI
CLO-4 :	Associate MPI blocking and non-blocking calls to send and receive data
CLO-5 :	Determine the performance of OpenMP and MPI on data structures like BFS, MST

Program Learning Outcomes (PLO)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-2 :	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-3 :	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-4 :	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-5 :	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-

Unit-1

Introduction, Need for parallel programming, Shared and distributed memory architecture ,Thread Basics, OpenMP programming model, Threads, Cores, Hyper-threading, OpenMP library functions, Parallel directives, OpenMP data environment

Unit-2

Reduction clause, Parallel loops, Work sharing ,Loop schedules, False sharing, Race conditions , Vector-vector operations

Unit-3

Synchronization, Critical section, Matrix-vector operations, MPI Basics, Building blocks of MPI, MPI Data Types, Groups

Unit-4

Communicators, MPI calls to send and receive data, Broadcasting calls, MPI gathering and scattering, MPI non-blocking calls, MPI reduction and all-to-all collectives

Unit-5

Introduction to parallel graph algorithms, BFS using OpenMP, Distributed BFS algorithm, OpenMP based MST, MPI based MST. , Case study

LAB EXPERIMENTS:

1. Program to compute rank and size of processes
2. OpenMP program to compute Pi value
3. Sum of an array elements using Parallel clause
4. Matrix addition using OpenMP
5. Program to illustrate critical section
6. Program to illustrate race conditions
7. Program to illustrate false sharing
8. Sum of elements using MPI
9. Matrix addition using MPI
10. MPI program using blocking calls
11. MPI program using non-blocking calls
12. Breadth First Search using OpenMP
13. Distributed BFS
14. *OpenMP based MST*
15. *MPI based MST*

Learning Resources	1. Parallel Programming in MPI and OpenMP The Art of HPC, volume 2 Victor Eijkhout 2nd edition 2022, formatted March 9, 2023 2. *Parallel Programming in C with MPI and OpenMP ,Michael J.Quinn,* Dubuque, Iowa : McGraw-Hill,2004.	3. Roman Trobec · Boštjan Slivnik Patricio Bulić · Borut Robič, Introduction to Parallel Computing From Algorithms to Programming on State-of-the-Art Platforms, 2018

Learning Assessment

	Bloom's Level of Thinking	*Continuous Learning Assessment (CLA)*				**Practical Exam (40% Weightage)**	
		Formative CLA-1 Average of unit test (45%)		*Life Long Learning CLA-2 – (15%)*			
		Theory	*Practice*	*Theory*	*Practice*	*Theory*	*Practice*
Level 1	*Remember*	*15%*	-	-	*15%*	*15%*	-
Level 2	*Understand*	*20%*	-	-	*20%*	*20%*	-
Level 3	*Apply*	*25%*	-	-	*25%*	*25%*	-
Level 4	*Analyze*	*25%*	-	-	*25%*	*25%*	-
Level 5	*Evaluate*	*10%*	-	-	*10%*	*10%*	-
Level 6	*Create*	*5%*	-	-	*5%*	*5%*	-
	Total	*100 %*		*100 %*		*100 %*	

Course Designers

Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Mr.Arockia Praveen,Mishus Workshop Pvt Ltd	1. Prof. Varalakshmi Perumal,Director,IST,CEG Campus,Anna University	1. Dr.S.Nagadevi, Assistant Professor,,SRMIST, Kattankulathur
2. V.SolaiMurugan, Joint Director, CDAC	2. Dr.S.Raghavan,Assistant Professor, IIITDM,Kancheepuram	2. Dr.R.Vidhya ,Assistant Professor,,SRMIST, Kattankulathur

Course Code	21HCSF003	Course Name	Fundamentals of Accelerated Computing with CUDA Using Python	Course Category	F	Foundation Course	L	T	P	C
							2	0	2	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil

Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	Recognize the basics of parallel programming
CLR-2 :	Illustrate the CUDA programming model
CLR-3 :	Identify the different memory handling in CUDA programming
CLR-4 :	Analyze the parallel perception of PyCuda and PyOpenCL
CLR-5 :	Implement CUDA libraries to solve real world problems

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Program Learning Outcomes (PLO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	Formulate the basics of parallel programming	3	2	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-2 :	Apply the CUDA programming for parallel algorithms	3	2	1	-	-	-	-	-	-	-	-	-	-	2	-
CLO-3 :	Identify the different memory handling in CUDA programming	3	3	-	-	2	-	-	-	-	-	-	-	-	2	-
CLO-4 :	Analyze the parallel perception of PyCuda and PyOpenCL	3	3	1	-	-	-	-	-	-	-	-	-	-	2	-
CLO-5 :	Implement CUDA libraries to solve real world problems	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-

Unit-1

Von Neumann Architecture, *Multi-node Computing, CPUs vs GPUs, GPGPU,GPU Components,, Parallel Programming Languages, Models, Directives and libraries, Speedups Serial/Parallel Code and problems, Types of Parallelism, NVIDIA GPU architecture*

Unit-2

NVIDIA AND CUDA, ALTERNATIVES TO CUDA, CUDA's Programming Model, Threads, Blocks, Grids, CUDA'S EXECUTION MODEL: STREAMING MULTIPROCESSORS AND WARPS, CUDA COMPILATION PROCESS

Unit-3

Parallel merging, Parallel reduction, Memory Handling with CUDA, CACHES, MEMORY HIERARCHY, LOCAL MEMORY/REGISTERS, REGISTER USAGE, SHARED MEMORY, CONSTANT MEMORY, GLOBAL MEMORY

Unit-4

Overview of PyCUDA, Scripting with PyCUDA, PyCUDA workflow, gpuarray, reduction kernel, Introduction to PyOpenCL,

Unit-5

CUDA Libraries with Scikit CUDA, Linear Algebra with CUBLAS, Fast Fourier Transforms with cuFFT, Fast Fourier transforms with cuFFT

LAB EXPERIMENTS:

1. Matrix –Matrix multiplication
2. Vector –Vector addition
3. Vector –vector multiplication
4. Matrix –Matrix addition
5. Compute PI value
6. Load balancing with CUDA
7. 1-dimensional tiling with CUDA
8. 2-d matricx multiplication
9. Single thread block vector addition
10. Multiple thread block vector addition
11. CUDA Atomic Reduction
12. PyCuda Program
13. PyOpenCL Program
14. Radix Sort
15. Profiling CUDA programs

Learning Resources	1. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012 2. David B. Kirk and Wen-mei W. Hwu, " Programming Massively Parallel Processors A Hands-on Approach" Second Edition, Elsevier Inc,2016 3. Prof. Stewart Weiss ,"GPUs and GPU Programming " Lecture Notes	4. https://on-demand.gputechconf.com/gtc/2010/presentations/S12041-PyCUDA-Simpler-GPU-Programming-Python.pdf 5. Dr.Brian Tuomanen,"Hands-on GPU Programming with Python and CUDA" 2018

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers

Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.Saju G Nair, Senior Development Manager, Kyndryl India Pvt. Ltd	Dr. Nileshchandra Pikle Assistant Professor&"A certified CUDA instructor by NVIDIA",IIIT, Nagpur	Dr.S.Nagadevi, Assistant Professor,SRMIST,Kattankulathur
Mr.Arockia Praveen, Mishus Workshop Pvt. Ltd		Dr.K.R.Jansi, Assistant Professor,SRMIST,Kattankulathur

Course Code	21HCSE001	Course Name	Advanced Computer Architecture	Course Category	E	Professional Elective	L	T	P	C
							3	0	0	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil

Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil

Course Learning Rationale :	The purpose of learning this course is to:
CLR-1 :	Understand the Fundamentals of computers, Memory operations and Addressing Modes
CLR-2 :	Know about Functions of Arithmetic and Logic unit
CLR-3 :	Explore the Operations of Control Unit, Execution of Instruction and Pipelining
CLR-4 :	Classify the Need for Parallelism, Multicore and Multiprocessor Systems
CLR-5 :	Understand the Concepts and functions of Memory unit, I/O unit

Course Outcomes (CO):	At the end of this course, learners will be able to:	Program Outcomes (PO)														
		1 Engineering Knowledge	2 Problem Analysis	3 Design & Development	4 Analysis, Design, Research	5 Modern Tool Usage	6 Society & Culture	7 Environment & Sustainability	8 Ethics	9 Individual &Team Work	10 Communication	11 Project Mgt. & Finance	12 Life Long Learning	13 PSO - 1	14 PSO - 2	15 PSO – 3
CO-1 :	Identify the computer hardware and how software interacts with computer hardware	3	2	-	-	-	-	-	-	-	-	-	-	1	-	-
CO-2 :	Apply Boolean algebra as related to designing computer logic ,through simple combinational and sequential logic circuits	3	2	-	-	-	-	-	-	-	-	-	-	-	2	-
CO-3 :	Examine the detailed operation of Basic Processing units and the performance of Pipelining	3	-	-	-	-	-	-	-	-	-	-	-	-	-	1
CO-4 :	Analyze concepts of parallelism and multi-core processors.	3	-	-	-	-	-	-	-	-	-	-	-	-	2	-
CO-5 :	Classify the memory technologies, input-output systems and evaluate the performance of memory system	3	2	-	-	-	-	-	-	-	-	-	-	-	3	-

Unit-1 Theory of Parallelism Theory of Parallelism: Parallel Computer Models, The State of Computing, Multiprocessors and Multicomputer ,Multivector and SIMD Computers ,PRAM and VLSI Models, Program and Network Properties ,Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures
Unit-2 Hardware Technologies Hardware Technologies: Processors and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology.
Unit-3 Bus, Cache, and Shared Memory Bus, Cache, and Shared Memory ,Bus Systems ,Cache Memory Organizations ,Shared Memory Organizations ,Sequential and Weak Consistency Models ,Pipelining and Superscalar Techniques ,Linear Pipeline Processors ,Nonlinear Pipeline Processors ,Instruction Pipeline Design ,Arithmetic Pipeline Design
Unit-4 Parallel and Scalable Architectures Parallel and Scalable Architectures: Multiprocessors and Multicomputers ,Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Three Generations of Multicomputers ,Message-Passing Mechanisms ,Multivector and SIMD Computers ,Vector Processing Principles ,Multivector Multiprocessors ,Compound Vector Processing ,SIMD Computer Organizations
Unit-5 Software for parallel programming Software for parallel programming: Parallel Models, Languages, and Compilers ,Parallel Programming Models, Parallel Languages and Compilers ,Dependence Analysis of Data Arrays ,Parallel Program Development and Environments, Synchronization and Multiprocessing Modes. Instruction and System Level Parallelism, Instruction Level Parallelism ,Computer Architecture ,Contents, Basic Design Issues ,Problem Definition ,Model of a Typical Processor ,Compiler-detected Instruction Level Parallelism ,Operand Forwarding ,Reorder Buffer, Register Renaming ,Tomasulo's Algorithm ,Branch Prediction, Limitations in Exploiting Instruction Level Parallelism ,Thread Level Parallelism.

Learning Resources	
	1. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015
	2. John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition, Morgan Kaufmann Elseveir, 2013

Bloom's Level of Thinking		Continuous Learning Assessment (60% weightage)		Final Examination (40% weightage)
		CLA – 1 (50%)	CLA – 2 (10%)	
		Theory	Theory	Theory
Level 1	Remember	*20%*	*20%*	*20%*
Level 2	Understand	*20%*	*20%*	*20%*
Level 3	Apply	*50%*	*50%*	*50%*
Level 4	Analyze	10%	10%	10%
Level 5	Evaluate			
Level 6	Create			
Total		100 %	100 %	100 %

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.Vimal,CDAC	Dr.Noor Mohammed, IIITDM,Kancheepuram	Dr.S.Nagadevi, Assistant Professor,SRMIST,Kattankulathur
Mrs.V.A.Prabha,CDAC	Dr.Mohammed Hazan, IIITDM,Kancheepuram	Dr.R.Jeya, Assistant Professor,SRMIST,Kattankulathur

Course Code	21HCSE002	Course Name	GPU Computing	Course Category	E	Professional Elective	L	T	P	C
							2	0	2	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	Understand the difference between different parallel programming architectures.
CLR-2 :	Knowledge of GPU aware programming using CUDA and OpenAcc frameworks.
CLR-3 :	Design and develop GPU accelerated real-world simulations and applications
CLR-4 :	Acquire the knowledge on OpenAcc
CLR-5 :	Demonstrate applications using CUDA programming

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Program Learning Outcomes (PLO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	Understand the difference between different parallel programming architectures.	3	2	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-2 :	Knowledge of GPU aware programming using CUDA and OpenAcc frameworks.	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-3 :	Design and develop GPU accelerated real-world simulations and applications	3	3	-	-	2	-	-	-	-	-	-	-	-	2	-
CLO-4 :	Acquire the knowledge on OpenAcc	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-5 :	Demonstrate applications using CUDA programming	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-

Unit-1
GPU ARCHITECTURE **9**
Trends in CPU and GPU performance – serial versus parallel programming and Flynn's Taxonomy-- -- History of GPUs Evolution of GPU architectures – Understanding Parallelism with GPU –Typical GPU Architecture – Heterogeneous Programming

Unit-2
CUDA OVERVIEW : CUDA Hardware Overview – Threads, Blocks, Grids, Warps, Scheduling – CUDA Memory Model – Built-in variables and functions – thread scheduling – thread synchronization --- vector addition -- matrix multiplication

Unit-3
CUDA PROGRAMMING **9**
Using CUDA – Multi GPU – Multi GPU Solutions – Optimizing CUDA Applications: Problem Decomposition, Memory Considerations, Transfers, Thread Usage, Resource Contentions.

Unit-4
OpenAcc PROGRAMMING **9**
OpenAcc Programming Cycle – Directives – data regions – data clauses – optimizing loop parallelism – linear algebra

Unit-5
GPU PERFORMANCE **9**
Parallel reduction – Memory Coalescing – loop unrolling – thread granularity -- Parallel Patterns: Convolution, Prefix Sum, Sparse Matrix – Matrix Multiplication – Programming Heterogeneous Cluster.

LAB EXPERIMENTS:

1. Matrix –Matrix multiplication
2. Vector –Vector addition
3. Vector –vector multiplication
4. Matrix –Matrix addition
5. Compute PI value
6. Load balancing with CUDA
7. 1-dimensional tiling with CUDA
8. 2-d matricx multiplication
9. 1D -Convolution
10. Single thread block vector addition
11. Multiple thread block vector addition
12. CUDA Atomic Reduction
13. Jacobi Iteration using OpenAcc
14. Loop parallelism using OpenAcc
15. Data locality using OPenAcc

Learning Resources	1. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012 2. https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html	3. Nicholas Wilt, ―CUDA Handbook: A Comprehensive Guide to GPU Programming, Addison – Wesley, 2013. 4. JASON SANDERS EDWARD KANDROT , CUDA by Example A general purpose , NVIDIA

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.V.Solai Muruganm, Joint Director, CDAC	Prof.Ananthakumar,MIT	Dr.S.Nagadevi,Assistant Professor,SRMIST,Kattankulathur
		Dr.D.Malathy,Professor, SRMIST,Kattankulathur

Course Code	21HCSE003	Course Name	Accelerated Deep Learning	Course Category	E	Professional Elective	L	T	P	C
							2	0	2	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	Recognize the basics of learning problems
CLR-2 :	Illustrate the decision tree algorithm for classification and prediction
CLR-3 :	Identify the probability based and instance-based learning methods to solve the real-world problems
CLR-4 :	Express the perception of neurons and network functioning
CLR-5 :	Demonstrate the working NN models to solve real world problems

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:
CLO-1 :	Formulate the Learning problems
CLO-2 :	Apply the decision tree algorithm for classification and prediction
CLO-3 :	Apply probability based and instance-based learning to solve the real-world problems
CLO-4 :	Analyze the perception of neurons and network functioning
CLO-5 :	Implement NN models to solve real world problems

Program Learning Outcomes (PLO)														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual & Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
3	2	-	-	-	-	-	-	-	-	-	-	-	2	-
3	3	-	-	-	-	-	-	-	-	-	-	-	2	-
3	3	-	-	2	-	-	-	-	-	-	-	-	2	-
3	3	-	-	-	-	-	-	-	-	-	-	-	2	-
3	3	-	-	-	-	-	-	-	-	-	-	-	2	-

Unit-1
Introduction to Deep Learning, Bayesian Learning, Decision Surfaces,Linear Classifiers, Linear Machines with Hinge Loss,Optimization Techniques, Gradient Descent, Batch Optimization.

Unit-2
Basics of Neural Networks:Basic Concept of Neurons – Perceptron Algorithm – Feed Forward and Backpropagation Networks,Unsupervised Learning with Deep Network, Autoencoders

Unit-3
Convolutional Neural Networks
CNN Architectures – Convolution – Pooling Layers – Transfer Learning – Image Classification using Transfer Learning – Recurrent and Recursive Nets – Recurrent Neural Networks – Deep Recurrent Networks – Recursive Neural Networks – Applications.

Unit-4
Revisiting Gradient Descent, Momentum Optimizer, RMSProp, Adam,Effective training in Deep Net- early stopping, Dropout, Batch Normalization, Instance Normalization, Group Normalization, Recent Trends in Deep Learning Architectures, Residual Network, Skip Connection Network, Fully Connected CNN

Unit-5
Applications of Deep Learning
Images segmentation – Object Detection – Automatic Image Captioning – Image generation with Generative adversarial networks – Video to Text with LSTM models – Attention models for Computer Vision – Case Study: Named Entity Recognition – Opinion Mining using Recurrent Neural Networks – Parsing and Sentiment Analysis using Recursive Neural Networks – Sentence Classification using Convolutional Neural Networks – Dialogue Generation with LSTMs.

LAB EXPERIMENTS:

1. Face Recognition
2. Iris Recognition,
3. Ear biometrics,
4. Bio traits Analysis ,
5. Lift Automation in University Building with crowd sourcing of Signals –Floor Navigator,
6. Indigenous 32 bit RISC processor for ASIC implementation,
7. Development of Algorithms for Large graphs;
8. Pre-fetching for GPUs,
9. Deploying AI Deep Learning Models with scalable classification problems,
10. Virtual Digital Assistants

Learning Resources	1. Ian J. Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2017. 2. Francois Chollet, "Deep Learning with Python", Manning Publications, 2018 3. Phil Kim, "Matlab Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence", Apress, 2017.	4. Ragav Venkatesan, Baoxin Li, "Convolutional Neural Networks in Visual Computing", CRC Press, 2018. 5. Navin Kumar Manaswi, "Deep Learning with Applications Using Python", Apress, 2018. 6. Joshua F. Wiley, "R Deep Learning Essentials", Packt Publications, 2016. 7. Pattern Classification- Richard O. Duda, Peter E. Hart, David G. Stork, John Wiley & Sons Inc.

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers

Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Mr.Arockia Praveen,Mishus Workshop Pvt Ltd	1. Prof. Varalakshmi Perumal,Director,IST,CEG Campus,Anna University	1. Dr.S.Nagadevi, Assistant Professor,,SRMIST, Kattankulathur
2. V.SolaiMurugan, Joint Director, CDAC	2. Dr.S.Raghavan,Assistant Professor, IIITDM,Kancheepuram	2. Dr.R.Vidhya ,Assistant Professor,,SRMIST, Kattankulathur

Course Code	21HCSE004	Course Name	Introduction to Concurrent Programming with GPUs	Course Category	E	Professional Elective	L	T	P	C
							2	0	2	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	To understand the power of parallelism on GPU
CLR-2 :	To understand the To illustrate the parallel programming with Python
CLR-3 :	To harness the power of parallel computing with NVIDIA GPUs
CLR-4 :	To understand and to apply CUDA over the massively parallel hardware
CLR-5 :	To execute applications using parallel programming models

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Program Learning Outcomes (PLO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	Exhibit knowledge on basics of Parallel programming using GPU	3	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-2 :	To illustrate the parallel programming with Python	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-3 :	Apply the parallel computing with NVIDIA GPUs	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-4 :	To understand and to apply CUDA over the massively parallel hardware	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-5 :	To execute applications using parallel programming models	1	2	3	-	-	-	-	-	-	-	-	-	-	-	-

Unit-1

Parallel Programming on CPU and GPU: INTRODUCTION, GPUs as Parallel Computers. Architecture of a Modern GPU, Parallel Programming Languages and Models, Evolution of Graphics Pipelines, GPU Computing,GPU Performance Trend,Multi-GPU Computing

Unit-2

Parallel Programming with Python: Multicore (local) Parallel Programming, Multiprocessing,Pool,Parallel map/reduce,Asynchronous Functions and Futures, Asynchronous Mapping,Multinode (distributed/cluster) Parallel Programming, Scoop, Distributed map/reduce,Running Scoop on a Cluster

Unit-3

Introduction to NVIDIA GPUs: GPU Computing Fundamentals, APPLYING AMDAHL'S LAW Math and Communication , cuBLAS , cuTENSOR , MULTI GPU MATH LIBRARIES ,Introduction to OpenACC, OpenACC COMPILER DIRECTIVES, OPENACC PARALLEL DIRECTIVE,Data Synchronization,Loop Optimization

Unit-4

Introduction to CUDA:Introduction to CUDA, Kernel call, Passing Parameters, CUDA parallel programming,summing vectors, Threads, blocks, vector sum, dot product, constant memory, matrix multiplication

Unit-5

CUDA Libraries with Scikit CUDA, Linear Algebra with CUBLAS, Fast Fourier Transforms with cuFFT, Fast Fourier transforms with cuFFT, CUDA Tools

LAB EXPERIMENTS:

1. Program to compute rank and size of processes
2. Matrix –Matrix multiplication
3. Vector –Vector addition
4. Vector –vector multiplication
5. Matrix –Matrix addition
6. Compute PI value
7. Load balancing with CUDA
8. Matrix Multiplication
9. Jacobi Iteration
10. Sum of an array
11. Program using Parallel construct
12. Program using reduction clause
13. Accelerate the loops with kernel construct
14. Offload Inefficient Operations to Maintain Data Locality
15. Program to Optimize Data Locality

Learning Resources	1. http://cscads.rice.edu/Owens.pdf 2. Programming Massively Parallel Processors: A Hands-on Approach Paperback – Import, 20 December 2012by David B. Kirk (Author), Wen-mei W. Hwu (Author) 3. https://chryswoods.com/parallel_python/	4. https://cades.ornl.gov/wp-content/uploads/2019/10/CADES-gpu-training-nov2019.pdf 5. JASON SANDERS EDWARD KANDROT ,CUDA by Example A general purpose , NVIDIA 6. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Mr.Blessin George Varghese,Bajaj Finserv	1. Dr.Noor Mohammed, IIITDM,Kancheepuram	1. Dr.S.Nagadevi,Assistant Professor,SRMIST,Kattankulathur
	2. Dr.Mohammed Hazan, IIITDM,Kancheepuram	2. Dr.K.R.Jansi,Assistant Professor,SRMIST Kattankulathur

Course Code	21HCSE005	Course Name	Applied Accelerated Artificial Intelligence	Course Category	E	Professional Elective	L	T	P	C
							2	0	2	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	*The purpose of learning this course is to:*
CLR-1 :	*Recognize the basics of learning problems*
CLR-2 :	*Illustrate the decision tree algorithm for classification and prediction*
CLR-3 :	*Identify the probability based and instance-based learning methods to solve the real-world problems*
CLR-4 :	*Express the perception of neurons and network functioning*
CLR-5 :	*Demonstrate the working NN models to solve real world problems*

Course Learning Outcomes (CLO):	*At the end of this course, learners will be able to:*	**Program Learning Outcomes (PLO)**														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual & Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	*Formulate the Learning problems*	*3*	*2*	-	-	-	-	-	-	-	-	-	-	-	*2*	-
CLO-2 :	*Apply the decision tree algorithm for classification and prediction*	*3*	*3*	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-3 :	*Apply probability based and instance-based learning to solve the real-world problems*	*3*	*3*	-	-	*2*	-	-	-	-	-	-	-	-	2	-
CLO-4 :	*Analyze the perception of neurons and network functioning*	*3*	*3*	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-5 :	*Implement NN models to solve real world problems*	*3*	*3*	-	-	-	-	-	-	-	-	-	-	-	2	-

Unit-1

Introduction to AI System Hardware CPU, RAM, GPU, Interconnects, Storage, Network Controller; Introduction to AI Accelerators GPUs ,Introduction to System Software Operating System, Virtualization, Cloud; Introduction to Containers and IDE ,Scheduling and Resource Management Introduction to schedulers and orchestration tools

Unit-2

DeepOps: Deep-dive into Kubernetes with deployment of various AI-based services , Design principles for building High Performance compute clusters for AI
Implementation details for building High Performance compute clusters for AI, Frameworks for Accelerated Deep Learning Workloads - PyTorch; Frameworks for Accelerated Deep Learning Workloads Accelerated PyTorch

Unit-3

TensorFlow ,Accelerated TensorFlow , Optimizing Deep Learning Training: Automated Mixed Precision ,Optimizing Deep Learning Training, Fundamentals of Distributed AI Computing: Multi-GPU and multi-node implementation (MPI, NCCL, RDMA) Distributed AI Computing: Horovod

Unit-4

Challenges with Distributed Deep Learning Training Convergence ,Fundamentals of Accelerating Deployment , Accelerating neural network inference in PyTorch and TensorFlow Accelerated Data Analytics ,Accelerated Machine Learning , Introduction to NLP,Text Classification using word embedding

Unit-5

Scale Out with DASK; Web visualizations to GPU accelerated crossfiltering ,Accelerated ETL Pipeline with SPARK, Applied AI: Smart City -Intelligent Video Analytics; Applied AI: Healthcare -Federated Learning, AI Assisted Annotation;

LAB EXPERIMENTS:

1. Face Recognition
2. Iris Recognition,
3. Ear biometrics,
4. Bio traits Analysis ,
5. Lift Automation in University Building with crowd sourcing of Signals –Floor Navigator,
6. Indigenous 32 bit RISC processor for ASIC implementation,
7. Development of Algorithms for Large graphs;
8. Pre-fetching for GPUs,
9. Deploying AI Deep Learning Models with scalable classification problems,
10. Virtual Digital Assistants

Learning Resources	1. Ashutosh Mishra (Editor), Jaekwang Cha (Editor), Hyunbin Park (Editor), Shiho Kim (Editor),Artificial Intelligence and Hardware Accelerators Hardcover – Import, 16 March 2023,Cham : Springer International Publishing : Imprint: Springer, 2023,Edition:1st ed. 2023. 2. https://doras.dcu.ie/20641/1/FICloud15-EdgeCloudContainer.pdf 3. https://github.com/NVIDIA/deepops/blob/master/docs/k8s-cluster/README.md 4. Programming PyTorch for Deep Learning,by Ian Pointer,Released September 2019,Publisher(s): O'Reilly Media, Inc.ISBN: 9781492045359	

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.Vimal,CDAC	Dr.Noor Mohammed, IIITDM,Kancheepuram	1. Dr.S.Nagadevi,Assistant Professor,SRMIST,Kattankulathur
Mrs.V.A.Prabha,CDAC	Dr.Mohammed Hazan, IIITDM,Kancheepuram	2. Dr.K.R.Jansi,Assistant Professor,SRMIST Kattankulathur

Course Code	21HCSE006	Course Name	Fundamentals of Accelerated Computing with OpenACC	Course Category	E	Professional Elective	L	T	P	C
							2	0	2	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	Recognize the basics of OpenACC accelerator model
CLR-2 :	Illustrate the Profile and optimize your CPU-only applications to identify hot spots for acceleration.
CLR-3 :	Identify the Use OpenACC directives to GPU-accelerate your codebase.
CLR-4 :	Express the perception of CUDA and its processes
CLR-5 :	Demonstrate thedata movement between the CPU and GPU accelerator.

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Program Learning Outcomes (PLO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	Recognize the basics of OpenACC accelerator model	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-2 :	Illustrate the Profile and optimize your CPU-only applications to identify hot spots for acceleration.	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-3 :	Identify the Use OpenACC directives to GPU-accelerate your codebase.	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-4 :	Express the perception of CUDA and its processes	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-
CLO-5 :	Demonstrate thedata movement between the CPU and GPU accelerator.	3	2	1	-	-	-	-	-	-	-	-	-	-	-	-

Unit-1
OpenACC, OpenACC Accelerator Model,Accelerating an Application with OpenACC , OpenACC Directive Syntax,Porting Cycle,Profiling

Unit-2
The Kernels Construct, Parallel Construct, The Loop Construct, Routine Directive, Atomic Operations, Data Regions, Data Clauses, Shaping Arrays, Unstructured Data Lifetimes, Update Directive, Optimize

Unit-3
Data Locality,Optimize Loops, 3 Levels of Parallelism, Mapping Parallelism to the Hardware , Collapse Clause, Routine Parallelism, Optimize Loops

Unit-4
OpenACC Interoperability, Host Data Region, Using Device Pointers, Asynchronous Queues and CUDA Streams, CUDA Managed Memory, Using CUDA Device Kernels

Unit-5
Asynchronous Operation,Multi-device Programming,Case Study: Asynchronous Pipelining of a Mandelbrot Set

LAB EXPERIMENTS:

1. Vector addition
2. Matrix addition
3. Matrix Multiplication
4. Jacobi Iteration
5. Sum of an array
6. Program using Parallel construct
7. Program using reduction clause
8. Accelerate the loops with kernel construct
9. Offload Inefficient Operations to Maintain Data Locality
10. Program to Optimize Data Locality
11. Obtaining Device and Host Pointer Addresses
12. Asynchronous Pipelining of a Mandelbrot Set
13. Program for Multi-device Programming
14. Program using collapse clause
15. Dot product

Learning Resources	1. OpenACC Programming and Best Practices Guide, June 2015, openacc-standard.org 2. OPENACC TUTORIAL GRIDKA SCHOOL 2018 30 August 2018 Andreas Herten Forschungszentrum Jülich 3. *https://ulhpc-tutorials.readthedocs.io/en/latest/gpu/openacc/basics/*	4. OpenACC for Programmers: Concepts and Strategies, First Editionby Guido Juckeland, Sunita Chandrasekaran Released September 2017 Publisher(s): Addison-Wesley Professional ISBN: 9780134694306

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers

Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Mr.Vimal,CDAC	3. Dr.Noor Mohammed, IIITDM,Kancheepuram	1. Dr.S.Nagadevi, Assistant Professor,,SRMIST, Kattankulathur
4. Mrs.V.A.Prabha,CDAC	5. Dr.Mohammed Hazan, IIITDM,Kancheepuram	2. Dr.R.Revathi ,Assistant Professor,,SRMIST, Kattankulathur

Course Code	21HCSE007	Course Name	Sparse Intensive GPU Computing	Course Category	E	Professional Elective	L	T	P	C
							2	0	0	2

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	To design an Indian Sparsity framework by creating supercomputing framework
CLR-2 :	To understand the design and implementation strategies of various applications of complex optimization problems
CLR-3 :	To harness the sparse computing facilities with new AI applications using incremental customization
CLR-4 :	To enable sparsity and to apply manipulations over the massively parallel hardware
CLR-5 :	To execute applications using sparse computation independently and parallel

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:
CLO-1 :	Exhibit knowledge on basics of Indian Sparsity framework
CLO-2 :	Identify and design and implementation strategies of various applications of complex optimization problems
CLO-3 :	Predict the type of security to be applied for various cloud services
CLO-4 :	Examine the concept of virtualization and capacity planning
CLO-5 :	Recommend the service provider for specific requirement

Program Learning Outcomes (PLO)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual & Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
2	1	-	-	-	-	-	-	-	-	-	-	-	-	-
1	3	1	-		-	-	-	-	-	-	-	-	-	-
1	-	2	-	-	-	-	-	-	-	-	-	-	-	-
2	1		-	2	-	-	-	-	-	-	-	-	-	-
2	-		1	2	-	-	-	-	-	-	-	-	-	-

Unit-1 **6 Hrs**

Overview on GPUs and Sparse Data:History of GPUs Evolution of GPU architectures – Understanding Parallelism with GPU- EDA – Data Processing -- Sparse Data-Sparse matrix Formats- Sparse data applications

Unit-2 **6 Hrs**

Sparse Matrices:Introduction- Graph Representations – Permutations and Re-orderings – Storage Schemes—Basic sparse matrix operations – Sparse Direct Solution Methods -- sparse matrix computation

Unit-3 **6 Hrs**

Sparse GPU Kernels for Deep Learning:Sparse matrices in deep learning -- dense–dense matrix multiplication -- Sparse Matrix-Vector Multiplication on GPUs --Sparse Matrix transpose Vector Multiplication on GPUs -- Matrix Factorization – Sparsity on GPUs

Unit-4 **6 Hrs**

Parallel Programming with CUDA:CUDA Hardware Overview – CUDA Programming -- Threads, Blocks, Grids, Warps, Scheduling – Memory Handling with CUDA: Shared Memory, Global Memory, Constant Memory and Texture Memory.

Unit-5 **6 Hrs**

Sparse Applications:Sparse Matrices: Sparse Decompositions and Ensembles -- Optimization of GPU based sparse matrix -- Recommendation Systems – Sparse neural networks

Learning Resources	1. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012 2. David B. Kirk and Wen-mei W. Hwu, " Programming Massively Parallel Processors A Hands-on Approach" Second Edition, Elsevier Inc,2016 3. Prof. Stewart Weiss ,"GPUs and GPU Programming " Lecture Notes 4. https://www.nvidia.com/docs/IO/66889/nvr-2008-004.pdf 5. https://cs.stanford.edu/~matei/papers/2020/sc_sparse_gpu.pdf	6. Sparse Matrix Formats Evaluation and Optimization on a GPU 7. https://developer.nvidia.com/gpugems/gpugems2/part-iv-general-purpose-computation-gpus-primer/chapter-33-implementing-efficient 8. https://www.diva-portal.org/smash/get/diva2:821031/FULLTEXT01.pdf 9.https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLVI-4-W3-2021/295/2022/isprs-archives-XLVI-4-W3-2021-295-2022.pdf 10. https://stanford.edu/~jduchi/projects/DuchiJoMc13_nips.pdf

Learning Assessment							
	Bloom's Level of Thinking	*Continuous Learning Assessment (CLA)*				*Summative Final Examination (40% weightage)*	
		Formative CLA-1 Average of unit test (50%)		*Life Long Learning CLA-2 – (10%)*			
		Theory	*Practice*	*Theory*	*Practice*	*Theory*	*Practice*
Level 1	*Remember*	*15%*	-	*15%*	-	*15%*	-
Level 2	*Understand*	*25%*	-	*20%*	-	*25%*	-
Level 3	*Apply*	*30%*	-	*25%*	-	*30%*	-
Level 4	*Analyze*	*30%*	-	*25%*	-	*30%*	-
Level 5	*Evaluate*	-	-	*10%*	-	-	-
Level 6	*Create*	-	-	*5%*	-	-	-
	Total	*100 %*		*100 %*		*100 %*	

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Mr.Hemant, NVIDIA	Dr. Nileshchandra Pikle Assistant Professor&"A certified CUDA instructor by NVIDIA",IIIT, Nagpur	1. Dr.B.Amutha,Professor,SRMIST
		2. Dr.S.Nagadevi,Assistant Professor,SRMIST

Course Code	21HCSE008	Course Name	NVIDIA GPU's Accelerated Computing	Course Category	E	Professional Elective	L	T	P	C
							2	0	1	2

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards		Nil	

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	To understand the power of parallelism and OpenMP framework
CLR-2 :	To understand the design and implementation strategies of various applications of complex problems using MPI
CLR-3 :	To harness the parallel computing facilities for numerical methodswith OpenMP and MPI
CLR-4 :	To understand and to apply OpenACC over the massively parallel hardware
CLR-5 :	To execute applications using parallel programming models

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Program Learning Outcomes (PLO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	Exhibit knowledge on basics of Parallel programming using OpenMP	3	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-2 :	To illustrate strategies of various applications of MPI programs	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-3 :	Apply the numerical methods in OpenMP and MPI	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-4 :	Examine the concept of OpenACC in parallelization	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CLO-5 :	Implement real time projects	1	2	3	-	-	-	-	-	-	-	-	-	-	-	-

Unit-1
OpenMP
Motivation and need for parallelization, Examples and applications in scientific computing, Parallel programming paradigms, Terminology. OpenMP Programming : Basics, scope of variables, parallel loop directives, scheduling, critical directives

Unit-2
MPI Programming
MPI Programming : Basics, point-to-point and collective communication, MPI derived data types, performance evaluation, advanced function calls.)

Unit-3
Numerical Methods with MPI and OpenMP
Numerical integration, explicit and implicit finite-differences, solution of system of linear equations, solution of partial differential equations

Unit-4
OpenACC
Motivation, Compute Constructs (Kernel, Parallel, Loop, Routine), Data Directives, Reductions, Atomics, Data Transfers, Asynchronous Processing, Multi-Device Programming.. Same applications as for OpenMP and MPI, with focus on optimizing for GPUs

Unit-5
Real-time Projects:
Face Recognition, Iris Recognition, Ear biometrics, Bio traits Analysis , Lift Automation in University Building with crowd sourcing of Signals –Floor Navigator, Indigenous 32 bit RISC processor for ASIC implementation, Development of Algorithms for Large graphs; Pre-fetching for GPUs, Deploying AI Deep Learning Models with scalable classification problems, Virtual Digital Assistants

LAB EXPERIMENTS:

1. Program to compute rank and size of processes
2. OpenMP program to compute Pi value
3. Sum of an array elements using Parallel clause
4. Matrix addition using OpenMP
5. Program to illustrate critical section
6. Program to illustrate race conditions
7. Program o illustrate false sharing
8. Sum of elements using MPI
9. Matrix addition using MPI
10. MPI program using blocking calls
11. MPI program using non-blocking calls
12. Breath First Search using OpenMP
13. Program using Parallel construct
14. Program using reduction clause
15. Accelerate the loops with kernel construct

Learning Resources	1. An Introduction to PARALLEL PROGRAMMING, Peter S. Pacheco, Morgan Kaufmann, 2011 2. Parallel programming in C with MPI and OpenMP, Michael Quinn, McGraw Hill Education, 2017 3. OpenACC for Programmers: Concepts and Strategies, Sunita Chandrasekaran, Guido Juckeland, Addison Wesley, 2017 4. Parallel Scientific Computing in C++ and MPI, George Em Karniadakis, and Robert Kirby II, Cambridge Universities Press, 2003	5. Using MPI, William Gropp, Ewing Lusk, Anthony Skjellum, The MIT Press, 2014 6. Using OpenMP, Barbara Chapman, Gabriele Jost, Ruud van der Pas, The MIT Press, 2008

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.Blessin George Varghese,Bajaj Finserv	Dr.Noor Mohammed, IIITDM,Kancheepuram	1. Dr.B.Amutha,Professor,SRMIST
	Dr.Mohammed Hazan, IIITDM,Kancheepuram	2. Dr.S.Nagadevi,Assistant Professor,SRMIST

Course Code	21HCSE009	Course Name	Introduction to Python Programming	Course Category	E	Professional Elective	L	T	P	C
							2	0	1	2

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computing Technologies	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:
CLR-1 :	To understand the basics of python programming and various datatypes
CLR-2 :	To gain knowledge on operators in python
CLR-3 :	To understand the basic building blocks of python
CLR-4 :	To understand the concept of functions and packages
CLR-5 :	To develop GUI using toolkit

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Program Learning Outcomes (PLO)														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual &Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	Understand the basics of python programming	3	2	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-2 :	Knowledge of operators in python	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-3 :	Understand the control statements in python	3	3	-	-	2	-	-	-	-	-	-	-	-	2	-
CLO-4 :	Understand parameters and arguments that are passed to a function or method	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-
CLO-5 :	Create GUI desktop application with Tkinter	3	3	-	-	-	-	-	-	-	-	-	-	-	2	-

Unit-1

Introduction to Python, Variables and Assignments, Expressions, Statements, Indentation, Input and Output, Datatypes, Numbers and Strings, List,, Sets, Tuples ,Dictionaries, Type Conversion

Unit-2

Operators in Python, Arithmetic operator, Comparison Operator, Assignment operators, Bitwise Operators, Logical Operators

Unit-3

Control Statements, if, If-else, if-elif-else statements, break, continue, pass statements,

Unit-4

Basics of Functions, Parameters and Arguments ,keyword arguments , default arguments, Variable-length arguments, Anonymous functions, working with multidimensional arrays ,pointers, creation of modules, import statement, python packages

Unit-5

Exception handling, Errors and Exceptions, try-finally clause, user defined exception. String pattern matching, multithreading, GUI Programming, button, canvas, check button, entry, frame, label, list box, menu button, menu, message, radio button, scale, scrollbar

LAB EXPERIMENTS:

1. Python Program for simple interest
2. Python Program to swap two elements in a list
3. Program using Tuples
4. Python program using sets
5. Merging two dictionaries
6. To check prime number using control statements
7. To Find factorial of a number using loop
8. Python program to add two matrices
9. Program to illustrate functions
10. Find numbers divisible by another number
11. Simple calculator using python functions
12. Date and Time using python
13. Exception handling
14. Python GUI Programming
15. Python GUI Programming

Learning Resources	1. Python Programming : An Introduction to Computer Science , 'John Zelle', Third Edition 2. *Python Crash Course : A hands on project based – Introduction to programming*	

	Bloom's Level of Thinking	Formative CLA – 1 Average of unit test (45%)		Life Long Learning CLA – 2 Practice (15%)		Summative Final Examination (40% Weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	15%	-	-	%	15%	-
Level 2	Understand	20%	-	-	30%	20%	-
Level 3	Apply	35%	-	-	35%	35%	-
Level 4	Analyze	30%	-	-	35%	30%	-
Level 5	Evaluate	%	-	-	%	%	-
Level 6	Create	%	-	-	%	%	-
	Total	100 %		100 %		100 %	

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.Vimal,CDAC	Dr.Noor Mohammed, IIITDM,Kancheepuram	Dr.R.Vidhya, AP, SRMIST
Mrs.V.A.Prabha,CDAC	Dr.Mohammed Hazan, IIITDM,Kancheepuram	Dr.J.Ramaprabha, AP, SRMIST