Course Specifications

Course Title:	**High Performance Computing (HPC)**
Course Code:	**332**
Program:	**Computer Science and Information Technologies**
Department:	**Computer Science and Information**
College:	**College of Science at Az Zulfi**
Institution:	**Majmaah University**

Table of Contents

A. Course Identification

1. Credit hours:
2. Course type **a.** University ☐ College ☒ Department ☐ Others ☐ **b.** Required ☐ Elective ☐
3. Level/year at which this course is offered:
4. Pre-requisites for this course (if any)**:**
5. Co-requisites for this course (if any)**:**

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**		
2	**Blended**		
3	**E-learning**		
4	**Correspondence**		
5	**Other**		

7. Actual Learning Hours (based on academic semester)

No	Activity	Learning Hours
Contact Hours		
1	**Lecture**	
2	**Laboratory/Studio**	
3	**Tutorial**	
4	**Others** (specify)	
	Total	
Other Learning Hours*		
1	**Study**	
2	**Assignments**	
3	**Library**	
4	**Projects/Research Essays/Theses**	
5	**Others** (specify)	
	Total	

* The length of time that a learner takes to complete learning activities that lead to achievement of course learning outcomes, such as study time, homework assignments, projects, preparing presentations, library times

B. Course Objectives and Learning Outcomes

1. Course Description

The High Performance Computing most generally refers to the practice of aggregating computing power in a way that delivers much higher performance than one could get out of a typical desktop computer or workstation in order to solve large problems in science, engineering, or business.
The main objective of this course is to provide students the design, analysis, and implementation, of high-performance computational science and engineering applications. Illustrate on advanced parallel algorithms and concurrent processing.

2. Course Main Objective

1	Introduce students to the types of high-performance and parallel computer systems
2	Efficiently use Appropriate programming languages for scientific computations
3	Estimate the performance in different implementations
4	Optimize the performance of programs.
5	Develop solutions of parallel computing problems as leads to high-performance computing

3. Course Learning Outcomes

	CLOs	Aligned PLOs
1	**Knowledge:**	
1.1	▪ be able to transform algorithms in the computational area to efficient programming code for modern computer architectures	
1.2	▪ Be able to design and implement complex databases schemas using ER diagrams, normalization, integrity constraints, and advanced database system features such as stored procedures and triggers.	
2	**Skills :**	
2.1	▪ Be able to write, organize and handle programs for scientific computations	
2.2	▪ To be able to evaluate the suitability of different HPC solutions to common problems found in Computational Science.	
2.3	▪ To be able to evaluate the potential benefits and pitfalls of Grid Computing.	
3	**Competence:**	
3.1	Work in a group and learn time management.	

CLOs		Aligned PLOs
3.2	Learn how to search for information through library and internet.	
3.3	Present a short report in a written form and orally using appropriate scientific language	

C. Course Content

No	List of Topics	Contact Hours
1	Parallel Processing Concepts (Levels of parallelism (instruction, transaction, task, thread, memory, function)	9
2	Parallel Programming: Processor Architecture, Interconnect, Communication, Memory Organization, and Programming Models in high performance computing architectures. Memory hierarchy and transaction specific memory design Thread Organization	12
3	Fundamental Design Issues in Parallel Computing: a) Synchronization b) Scheduling c) Job Allocation d) Job Partitioning e) Dependency Analysis f) Mapping Parallel Algorithms onto Parallel Architectures g) Performance Analysis of Parallel Algorithms	12
4	Fundamental Limitations Facing Parallel Computing: a) Bandwidth Limitations b) Latency Limitations c) Latency Hiding/Tolerating Techniques and their limitations	9
Total		

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge**		
1.1	be able to transform algorithms in the computational area to efficient programming code for modern computer architectures	Lectures Lab demonstrations Case studies Individual presentations	Written Exam Homework assignments Class Activities Quizzes
1.2	Be able to design and implement complex databases schemas using ER diagrams, normalization, integrity constraints, and advanced database system features such as stored procedures and triggers.		
2.0	**Skills**		
2.1	Be able to write, organize and handle programs for scientific computations	Lectures Lab	Written Exam Homework
2.2	To be able to evaluate the suitability of		

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
	different HPC solutions to common problems found in Computational Science.	demonstrations Case studies Individual presentations	assignments Class Activities Quizzes
2.3	To be able to evaluate the potential benefits and pitfalls of Grid Computing.		
3.0	**Competence**		
3.1	Work in a group and learn time management	□ Exercises □ Problem solving □ oral quizzes □ Essay questions Encourage students to Implement a real wireless computing system.	□ Write reports □ Exercises related to specific topics
3.2	Learn how to search for information through library and internet.		
3.3	Present a short report in a written form and orally using appropriate scientific language		

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	First written mid-term exam	6	20%
2	Second written mid-term exam	12	20%
3	Presentation, class activities, lab activity, and group discussion	Every week	10%
4	Homework assignments	After every chapter	10%
5	Final written exam	15	40%
TOTAL			100%

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :

1. A total of 6 office hours per week in the lecturer schedule in order to facilitate the student.
2. Contacting students using e-mail , mobile, office telephone and website

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	Czarnul, P. (2018). Parallel Programming for Modern High Performance Computing Systems. Chapman and Hall/CRC.
Essential References Materials	Pinedo, M. (2012). Scheduling (Vol. 29). New York: Springer.

Electronic Materials	
Other Learning Materials	Video and presentations that are available with the instructor

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	Classrooms and, Library, as those are available at the college of science at Azzulfi
Technology Resources (AV, data show, Smart Board, software, etc.)	Smart Board
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	None

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of Teaching	Students	• Analysis of students' results. Observation during class work. • Students' evaluations. • Colleagues' evaluations. • Evaluation questionnaire filled by the students. Interview a sample of students enrolled in the course to take their opinions
Evaluation of Teaching	Program leaders	• Self-assessment. • External evaluation. Periodic review of course (the Commission of study plans)

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	
Reference No.	
Date	