SULTAN QABOOS UNIVERSITY

COURSE OUTLINE

PROGRAM: B.Sc. in Computer Science

1. Course Code	COMP5204	
2. Course Title	COMPUTER SCIENCE SPECIAL TOPICS (1) - Parallel Programming	
3. Credits	3	
4. Pre-requisite Course(s)	COMP3203 (Algorithms) + COMP4501 (OS) + COMP4502 (Networks)	
5. Co-requisite Course(s)	-	
6. Equivalent Course(s)	-	
7. Incompatible Course(s)	-	
8. Course Category	☐ University Requirement	☐ University Elective
	☐ College Requirement	☐ College Elective
	☐ Department Requirement	☒ Department Elective
	☐ Specialization Requirement	☐ Specialization Elective
	☐ Other (specify):	
9. Course Owner	College: Science	Department: Computer Science
10. Course Type	☒ Lecture	☐ Lecture/Lab
	☐ Lecture/Seminar	☐ Lecture/Studio
	☐ Lecture/Tutorial	☐ Lecture/Lab/Tutorial or Seminar
	☐ Tutorial	☐ Laboratory (Practical)
	☐ Field or Work Placement	☐ Studio
	☐ Seminar	☐ Internship
	☐ Workshop	☐ Project
11. Language of Instruction	English	

12. Course Description

This is a special topics course on parallel programming. It teaches how to design and implement efficient parallel programs. It covers the following topics: Parallel Computers, Message Passing Computing; Partitioning and Divide-And-Conquer Strategies; Pipelined Computations; Synchronous Computations; Load Balancing and Termination Detection; Programming with Shared Memory; Selected Parallel Programming Applications as student projects such as: Sorting, Numerical Algorithms, Image Processing, Searching, and Optimization.

13. Teaching/Learning Strategies

Lectures will include PowerPoint presentations as well as explanations with examples on the white board and class discussions. Students will practice writing and analyzing parallel programs through homework assignments and through a term project. The term project includes parallel program design, complexity analysis, implementaion and experimental evaluation as well as a written project report and an oral presentation.

14. Assessment Components and Weight [%]

☐ Quizzes	☐ Practical	☐ Other (specify):
☒ Homework assignments 20%	☐ Project	
☒ In-term examination(s) 40%	☒ Final examination 40%	

15. Grading Method

☒ A-F Scale ☐ Pass/Not passed

16. Textbook(s) and Supplemental Material

Parallel Programming, by Barry Wilkinson and Michael Allen, 2nd Edition, Prentice Hall

17. Matching Course Objectives with Program Outcomes and SQU Graduate Attributes

SQU Graduate Attributes

A. SQU graduates should be able to:	**B. SQU graduates possess**	**C. SQU graduates should**
1. apply the knowledge and skills relevant to the specialization 2. communicate effectively and use information and communication technologies 3. critically analyze complex information and present it in simple clear manner	1. interpersonal communication skills and alignment with culture of international labour market to assist them in practical life and in living successfully 2. skills and motivation for independent learning and engagement in lifelong learning and research 3. work ethics and positive values, and intellectual independence and autonomy 4. teamwork skills and display potential leadership qualities	relish good citizenship qualities, be conscious of their national identity and be socially responsible, engage in community affairs and be mindful of contemporary issues.

#	Intended Student Learning Outcome /Course Learning Objective	Relevant Program Outcome(s)	Applicable Attribute(s)
1.	Define terminology commonly used in parallel computing.	a1	A1
2.	Describe different parallel architectures, interconnection networks, and parallel programming models.	a1	A1
3.	Design an efficient parallel algorithm to solve a given computational problem.	b1, b2, c1, k1	A1, A3
4.	Analyze the running time complexity of a parallel algorithm as a function of the problem size and the number of processors.	a2, j1, j2	A1, A3
5.	Learn appropriate parallel programming tools..	h2	B2
6.	Use appropriate parallel programming tools to implement a parallel algorithm.	c2, i1, i2, k2	A1
7.	Evaluate experimentally the performance of a parallel implementation.	c3	A1
8.			
9.			
10.			
11.			
12.			
13.			
14.			
15.			
16.			
17.			
18.			
19.			
20.			

16. Student Responsibilities

It is the student's responsibility to know and comply with all University Academic Regulations relevant to participation in this course. These regulations specifically include attendance requirement and students` academic code of conduct.

For attendance, it is the student's responsibility to be punctual and to attend all classes.

Students are expected to perform their work with honesty and avoid any academic misconduct, which is defined as the use of any dishonest or deceitful means to gain some academic advantage or benefit. This can take many forms, including but not limited to, the following: copying, plagiarism, collusion and forging documents. For full details, please refer to the Undergraduate Academic Regulations and to the Student Academic Misconduct Policy.

Additionally, this course requires that you:

Course Assement:

Item	Date Out	Due Date	Weights
Homework Assignment 1	Week 3	Week 5	5%
Homework Assignment 2	Week 5	Week 7	5%
Midterm Exam 1	Week 8	Week 8	20%
Homework Assignment 3	Week 9	Week 11	5%
Homework Assignment 4	Week 11	Week 13	5%
Midterm Exam 2	Week 14	Week 14	20%
Final Exam	SUN 31 December 2017, 08:00 - 11:00		40%

Copying Policy: A zero mark will be assigned the first time a student is caught involved in copying and his/her name will be added to a watch list maintained by the Head of Department. Further repeated involvement in copying will cause the student to get an F grade. Other plagiarism penalties may apply as per the University regulations.

Late Submission Policy: no late submission beyond set due date

Useful Links:

Parallel Computing Tutorial: https://computing.llnl.gov/tutorials/parallel_comp/
MPI Tutorial: https://computing.llnl.gov/tutorials/mpi/
MPI forum and standard: http://www.mpi-forum.org/
MPICH: http://www.mcs.anl.gov/research/projects/mpich2/
Pthreads Tutorial: https://computing.llnl.gov/tutorials/pthreads/
OpenMP Tutorial: https://computing.llnl.gov/tutorials/openMP
CUDA C: http://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf/
OpenCL: http://www.nvidia.com/content/cudazone/download/OpenCL/NVIDIA_OpenCL_ProgrammingGuide.pdf

COURSE INFORMATION			
Course Code	COMP6195	**Course Title**	Advanced Special Topics - Parallel Programming
Semester/ Year	Fall 2017	**Section**	10
Day, Time, and Place	SUN & TUE: 14:15 - 15:35 CTM/D05		

Course Coordinator	Prof. Khaled Day		
Office Location	Room 0007	**Office Hours**	SUN, TUE: 11:00 - 12:00
Office Tel. Ext.	2231	**Email**	kday@squ.edu.om

Tentative Schedule

Week	Lecture/ Topic	Material to be covered	Assessment
1	1, 2	Parallel Computers	assignments, midterm, final
2	1, 2	Message-Passing Parallel Computing	assignments, midterm, final
3	1, 2	Message-Passing Parallel Computing	assignments, midterm, final
4	1, 2	Partitioning and Divide-and-Conquer Strategies	assignments, midterm, final
5	1, 2	Partitioning and Divide-and-Conquer Strategies	assignments, midterm, final
6	1, 2	Pipelined Computations	assignments, midterm, final
7	1, 2	Pipelined Computations	assignments, midterm, final
8	1	Midterm Exam Synchronous Computations	 assignments, final
9	2	Synchronous Computations	assignments, final
10	1, 2	Load Balancing and Termination Detection	assignments, final
11	1, 2	Load Balancing and Termination Detection	assignments, final
12	1, 2	Programming with Shared Memory	assignments, final
13	1, 2	Programming with Shared Memory	assignments, final
14	1, 2	Parallel Programming Applications	final
15	1, 2	Parallel Programming Applications	final

APPENDIX A: INSTRUCTORS OF MULTIPLE SECTIONS					
Section	**Instructor**	**Day, Time, and Place**	**Office Location and Extension**	**Email**	**Office Hours**

APPENDIX B: ADDITIONAL INFORMATION
Relevant Program Outcomes:

a1. An ability to apply knowledge of computing appropriate to the discipline.

a2. An ability to apply knowledge of mathematics appropriate to the discipline.

b1. An ability to analyze a problem.

b2. An ability to identify and define computing requirements of the solution.

c1. An ability to design a computer-based system, process, component, or program to meet identified requirements.

c2. An ability to implement a designed computer-based system, process, component, or program to meet identified requirements.

c3. An ability to evaluate an implemented computer-based system, process, component, or program to meet identified requirements.

h1. Recognition of the need for continuing professional development.

h2. Ability to engage in continuing professional development.

i1. An ability to use current techniques and skills necessary for computing practice.

i2. An ability to use current tools necessary for computing practice.

j1. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems.

j2. Demonstrate comprehension of the tradeoffs involved in design choices when modeling and designing computer-based systems.

k1. An ability to apply software design principles in the construction of software systems.

k2. An ability to use software development and programming tools in the construction of software systems.