Course Outline: Message-passing Programming

School	School of Informatics (EPCC)	College	College of Science and Engineering
Credit level (Normal year taken)	SCQF Level 11 (Postgraduate)	Availability	Not available to visiting students
SCQF Credits	10	ECTS Credits	5

Summary	*This course is delivered online for students on Online Learning programmes within the College of Science and Engineering only. On-campus students interested in the material should refer to INFR11163 – Message-passing Programming* Parallel programming by definition involves co-operation between processors to solve a common problem. The programmer has to define the tasks that will be executed by the processors, and also how these tasks exchange data and synchronise with each other. In the message-passing model the tasks are separate processes that communicate by explicitly sending each other messages. All parallel operations are performed via calls to some message-passing interface that is entirely responsible for interfacing with the physical communication network. This course uses the de facto standard for message passing, the Message Passing Interface (MPI), which is a library callable from C, C++ or Fortran. Parallel programs written using MPI can run on almost any system from a multicore laptop up to the world's largest supercomputers.
Course description	The course makes use of lecture content delivered in both video form and delivered live via e.g. Blackboard Collaborate. Practical activities will be set for students to undertake in their own time in advance of regular tutorial sessions run via Blackboard Collaborate, with discussion and support available from other students, demonstrators, and staff via dedicated discussion areas (e.g. Learn Discussion Boards, Slack). The course will cover the following topics: - The message-passing model - Message-passing parallelisation of a regular domain code - MPI terminology - The anatomy of send and receive (synchronous and asynchronous) - Point-to-point message-passing examples - Non-blocking operations - Collectives - Communicator management - Derived datatypes (focusing mainly on array subsections) - Practicalities / Hints and Tips

Entry Requirements (not applicable to Visiting Students)

Pre-requisites		Co-requisites	INFR11184 – Practical Introduction to High Performance Computing

Prohibited Combinations	INFR11163 – Message-passing Programming	Other requirements	Ability to program in C, C++ or Fortran.

Course Delivery Information

Academic year 2019/20, Not available to visiting students (SS1)	
Course Start	Semester 2
Timetable	Timetable
Learning and Teaching activities (Further Info)	Total Hours: 100 (Lecture Hours 14, Seminar/Tutorial Hours 6, Online Activities 10, Programme Level Learning and Teaching Hours 2, Directed Learning and Independent Learning Hours 68)
Additional Information (Learning and Teaching)	Please contact the School for further information
Assessment (Further Info)	Written Exam 0 %, Coursework 100 %, Practical Exam 0 %
Additional Information (Assessment)	This is a practical course and the associated programming exercises are at least as important as the lectures. The course is assessed by coursework which requires students to write an MPI program to solve a given problem and produce a report covering the program's design, implementation and performance, demonstrating scientific writing.
Feedback	Provided on assessed work and through discussion of practical exercises via scheduled Blackboard Collaborate class sessions, class-discussion, and
No Exam Information	

Learning Outcomes

On completion of this course, the student will be able to:

1. Describe the message-passing model in detail.

2. Explain the circumstances which cause issues such as deadlock.
3. Implement standard message-passing algorithms in MPI.
4. Measure and comment on the performance of MPI programs.
5. Design, implement and debug efficient parallel programs to solve regular-grid problems.

Reading List
“Using MPI: Portable Parallel Programming with the Message-Passing Interface”, William Gropp, Ewing Lusk and Anthony Skjellum.

Additional Information	
Graduate Attributes and Skills	Project planning. Project management skills. Effective written and diagrammatic communication. Technical writing. Data collection and analysis.
Keywords	Programming, message passing, parallel, distributed memory, MPI, HPC

Contacts			
Course organiser	Dr David Henty **Tel:** (0131 6)50 5960 **Email:** d.henty@epcc.ed.ac.uk	Course secretary	Mr Ben Morse **Tel:** (0131 6)51 3398 **Email:** Ben.Morse@ed.ac.uk