| | | | |
|---|---|---|---|
| Course Outline – Threaded Programming | | | |
| School | School of Informatics | 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 INFR11178 – Threaded Programming*<br><br>This course is a practical introduction to parallel programming using the threading model, which is commonly used on shared memory and multicore hardware. The majority of the course is focused on teaching the use of the industry standard OpenMP API. | | |
| Course description | After taking this course students should have a good practical understanding of multithreaded programming and be competent OpenMP programmers.<br><br>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).<br><br>The course will cover the following topics:<br>- Basic concepts of shared memory: threads, tasks, shared/private data, synchronisation.<br>- Concepts of OpenMP: parallel regions, shared/private variables, parallel loops, reductions<br>- OpenMP parallel regions and associated clauses<br>- OpenMP worksharing directives, scheduling of parallel loops<br>- OpenMP synchronisation: barriers, critical sections, atomics, locks.<br>- OpenMP tasks<br>- Additional features of OpenMP: nesting, orphaning, threadprivate globals, OpenMP 4.0 features<br>- OpenMP implementations<br>- Memory models.<br><br>Lectures will be followed by a practical examples illustrating the key concepts. Students will have the choice of using either C or Fortran in the practical programming sessions on OpenMP. | | |

| | | | |
|---|---|---|---|
| Entry Requirements (not applicable to Visiting Students) | | | |
| Pre-requisites | | Co-requisites | INFR11184 – Practical Introduction to High Performance Computing |

| | | | |
|---|---|---|---|
| Prohibited Combinations | INFR11178 – Threaded Programming | Other requirements | Students are required to be familiar with C or Fortran for practical exercises. |

| | |
|---|---|
| Course Delivery Information | |
| | |
| Academic year 2019/20, Not available to visiting students (SS1) | Quota: |
| 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 Course Organiser for further information |
| Assessment (Further Info) | Written Exam 0 %, Coursework 100 %, Practical Exam 0 % |
| Additional Information (Assessment) | 100% Coursework, split into two assignments. Feedback will be given on the first assignment before students are expected to start work on the second |
| Feedback | Students receive detailed individual text-based feedback on all components of the coursework. The coursework is split into two parts, with feedback given on the first part before students are expected to start work on the second part. Students are very welcome contact the course team to discuss the feedback. |
| No Exam Information | |
| | |

| |
|---|
| Learning Outcomes |
| On completion of this course, the student will be able to:<br>Understand the key concepts of multithreaded programming<br>Have a knowledge of the syntax and semantics of the OpenMP API.<br>Be able to write a correct OpenMP program in C/C++ or Fortran.<br>Have an appreciation of the advanced features of OpenMP<br>Understand key factors affecting performance of threaded programs |

| |
|---|
| Reading List |
| https://mitpress.mit.edu/books/using-openmp<br>https://mitpress.mit.edu/books/using-openmp-next-step |

| |
|---|
| Additional Information |

| | |
|---|---|
| Graduate Attributes and Skills | Effective written and diagrammatic communication.<br>Technical writing.<br>Data collection and analysis.<br>Reflection on learning and practice.<br>Adaptation to circumstances.<br>Solution Exploration, Evaluation and Prioritisation. |
| Keywords | Programming, threads, parallel, shared memory, OpenMP, HPC |

| Contacts | | | |
|---|---|---|---|
| Course organiser | Dr Mark Bull<br>**Tel:** (0131 6)50 6717<br>**Email:** m.bull@epcc.ed.ac.uk | Course secretary | Mr Ben Morse<br>**Tel:** (0131 6)51 3398<br>**Email:** Ben.Morse@ed.ac.uk |