## Contents

1 Overview ................................................. 3
   1.1 Prerequisites ........................................ 3

2 Plagiarism policy ...................................... 3

3 Course Website ......................................... 3

4 Instructors .............................................. 3
   4.1 Course Coordinator and Lecturer ..................... 3
   4.2 Interaction with the Instructors ..................... 4

5 Study Material ......................................... 4

6 Contact Sessions ....................................... 4
   6.1 Lecture Schedule ..................................... 4

7 Assessment .............................................. 4
   7.1 Assignments .......................................... 5
   7.2 Examination .......................................... 5
   7.3 Absence from an Assessment ......................... 5
   7.4 Calculation of Marks ................................ 5
   7.5 Conditions .......................................... 5
      7.5.1 Examination Entrance ........................... 5
      7.5.2 Result Criteria .................................. 6

8 Module Outline ......................................... 6
   8.1 Lecture Schedule ..................................... 6
1 Overview

Computational science relies on the analysis of often complex models for its empirical data and analyses typically involve an enormous amount of calculations. Parallel computing is one means of reducing the time needed to complete such calculations. This module will examine the kinds of problems that lend themselves to parallel computation and the methods for implementing programs to solve such problems. The aim of the module is to provide a background for parallel and distributed computing as well as practical knowledge of the implementation of computational experiments.

1.1 Prerequisites

There are no prerequisites for this module. However, the module contains a strong programming component and you will be required to write programs using either C or C++.

2 Plagiarism policy

This department considers plagiarism as a serious offense. Disciplinary action will be taken against a student who commits plagiarism. For more information about the university’s policy and for additional resources about plagiarism, please see http://www.ais.up.ac.za/plagiarism/index.htm

3 Course Website

This course shall make use of the Computer Science portal at: http://www.cs.up.ac.za/
The course website is hosted at: http://www.cs.up.ac.za/courses/COS786.

4 Instructors

This course will be coordinated and presented by one lecturer, whose contact details follow.

4.1 Course Coordinator and Lecturer

Mr Ronald Klazar
E-mail: rklazar@cs.up.ac.za
4.2 Interaction with the Instructors

Please note that your lecturer is appointed from outside the university to present this module. The lecturer is not resident in an office at the university and, possibly like you, is working during the day. Therefore, you are encouraged to make the most of the scheduled lectures to discuss the subject with the lecturer. You are also welcome to compose detailed emails to the lecturer in cases where you would like assistance with a complicated topic. Should you require a meeting, you may nevertheless request an appointment with your lecturer.

5 Study Material

There is no prescribed text for this course, as the topics covered are derived from a variety of sources. Therefore, references to source material will be made available during the semester.

6 Contact Sessions

The course is presented in one, two-hour lecture per week. While the lecture slides will be made available, it is ill-advised to base your notes on the lecture slides. It will be your responsibility to take adequate notes during lectures and to study the recommended reading material in order to substantiate the lectures.

6.1 Lecture Schedule

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Day</th>
<th>Time</th>
<th>Venue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tuesday</td>
<td>18:30-20:30</td>
<td>IT 4-3</td>
</tr>
</tbody>
</table>

7 Assessment

Your progress in this course will be evaluated by means of practical assignments that apply the theory introduced during lectures. You will be required to write one, extensive exam at the end of the course to complete your evaluation.
7.1 Assignments
 Assignments will be specified during the semester, as new work is completed in the lectures. Assignments may take the form of programming projects or short research reports. All assignments will contribute equally toward the semester mark.

7.2 Examination
 A three-day, exam will be written at the end of the course. This exam will be comprised of theoretical questions as well as short programming assignments.

7.3 Absence from an Assessment
 Please note that assignments submitted after the due date will not be accepted under any circumstances.

There are no Supplementary or Aegrotat Examinations provided on postgraduate level.

7.4 Calculation of Marks
 The final mark, $FM$, shall be calculated as follows:

$$FM = 0.6SM + 0.4EM$$

where $SM$ is the semester mark, composed wholly of practical assignment marks, and $EM$ is the examination mark.

7.5 Conditions
 You must gain Examination Entrance in order to write the exam. If you fail to obtain Examination Entrance, you will fail the course. In addition to gaining Examination Entrance, you will need to meet specific Result Criteria in order to pass the course.

7.5.1 Examination Entrance
 In order to obtain Examination Entrance you must achieve a Semester Mark of at least 40%.
7.5.2 Result Criteria

In order to pass the course you must achieve Examination Entrance as well as both of the following minimum results:
   Examination  40%
   Final Mark    50%

8 Module Outline

8.1 Lecture Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>28-07</td>
<td>MPI: Communication</td>
</tr>
<tr>
<td>2</td>
<td>04-08</td>
<td>MPI: Groups, Contexts, Communicators</td>
</tr>
<tr>
<td>3</td>
<td>11-08</td>
<td>MPI: Environment, Errors, Profiling</td>
</tr>
<tr>
<td>4</td>
<td>18-08</td>
<td>MPI: Process Creation and Management</td>
</tr>
<tr>
<td>5</td>
<td>25-08</td>
<td>MPI: Establishing Communication</td>
</tr>
<tr>
<td>6</td>
<td>01-09</td>
<td>MPI: Wave Algorithms</td>
</tr>
<tr>
<td>7</td>
<td>08-09</td>
<td>MPI: Traversal Algorithms</td>
</tr>
<tr>
<td>8</td>
<td>15-09</td>
<td>MPI: Election Algorithms</td>
</tr>
<tr>
<td>9</td>
<td>22-09</td>
<td>No Lecture</td>
</tr>
<tr>
<td>10</td>
<td>29-09</td>
<td>Termination Detection</td>
</tr>
<tr>
<td>11</td>
<td>06-10</td>
<td>Recess</td>
</tr>
<tr>
<td>12</td>
<td>13-10</td>
<td>Synchronous Networks</td>
</tr>
<tr>
<td>13</td>
<td>20-10</td>
<td>Crash Failures</td>
</tr>
<tr>
<td>14</td>
<td>27-10</td>
<td>T.B.A.</td>
</tr>
<tr>
<td>15</td>
<td>03-11</td>
<td>T.B.A.</td>
</tr>
</tbody>
</table>