Module Title       Date of Approval
LANGUAGE DESIGN and IMPLEMENTATION   November 2011

Module Code 6CC509
Application

Pre-requisite
Development

Module Level 6

Credit value 20

Total Number of Learning Hours 200

Key Words
Computer Science

Module Delivery
Blended / Face to Face

Module Description
Computer languages are a cornerstone of practical and theoretical applications of computer science. Therefore, this module provides an in-depth and pragmatically-focused exploration of the theory, concepts, paradigms, current research, and practical implementation issues involved in designing and implementing mark-up languages, special-purpose languages, and programming languages.

Module Learning
On successful completion of the module, students will be able to:

Outcomes

1. Demonstrate a critical awareness of the theory and practical issues involved in implementing language parsers, compilers and interpreters.

2. Design a computer language, and be able to implement significant portions of an interpreter and/or compiler for it.

Module Content

• Programming paradigms, in particular as they related to language design and implementation
    ○ Computational paradigms
        ▪ Imperative
        ▪ Functional
        ▪ Logical
    ○ Language paradigms, e.g.:
- Procedural
- Functional
- Logical
- Declarative
- Object-oriented

- Language design
- Language implementation
  - Parsing
  - Grammars
  - Optimisation
  - Compiler construction
  - Cross-compilation
  - Virtual and real machines
- Future developments and current research

Module Learning and Teaching Methods

Seminars allow considerable flexibility for delivery, permitting lectures, discussion sessions, and practical activities as appropriate. Drop-in clinics are intended to provide students an opportunity to discuss issues with the instructor, and for those weak in the technical aspects of the subject to obtain assistance.

<table>
<thead>
<tr>
<th>Activity Type</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seminars</td>
<td>24</td>
</tr>
<tr>
<td>Clinics</td>
<td>24</td>
</tr>
<tr>
<td>Guided independent study</td>
<td>152</td>
</tr>
</tbody>
</table>

Scheduled learning and teaching activities: 24%
Guided independent study: 76%

Module Assessment
Mode: 100% coursework

Coursework 1: 40% weighting
Students will produce an approximately 1500 word research paper related to cutting-edge developments in language design and/or implementation.

Coursework 2: 60% weighting
Students will design a language and present a working implementation of at least a significant portion of it. Project milestones will be used to monitor progress and provide formative feedback. This is equivalent to approximately 2500 words.

Reading list
Key texts
- Watt (2004), Programming Language Design Concepts, John Wiley & Sons
Essential texts


Other recommended reading

- Students are required to do wider reading from a variety of sources, particularly academic journals such as the Association of Computing Machinery publications accessible via the ACM Digital Library at http://acm.org/dl