| | |
|---|---|
| **Module Code** | CS3021 CS3421 |
| **Module Name** | Computer Architecture II |
| **Module Short Title** | n/a |
| **ECTS weighting** | 5 |
| **Semester/term taught** | Semester 1 |
| **Contact Hours** | Lecture hours: 27<br>Lab hours: 0<br>Tutorial hours: 6<br><br>Total hours: 33 |
| **Module Personnel** | Dr Jeremy Jones |
| **Learning Outcomes** | Students who have successfully completed this module should be able to:<br>1. Write simple IA32 and x64 assembly language functions<br>2. Explain the IA32 and x64 procedure calling conventions<br>3. Write programs that mix C/C++ and IA32 or x64 assembly language functions<br>4. Describe the RISC design philosophy and translate simple high level language programs into RISC-I assembly language<br>5. Explain the key concepts behind instruction level pipelining and know how to apply a number of techniques to overcome data, load and control hazards<br>6. Explain the advantages of using virtual memory, show how virtual addresses are mapped to physical addresses and demonstrate how the functionality of a MMU is integrated into an operating system<br>7. Explain the use of a memory hierarchy to reduce effective memory access times, describe the organisation and operation of a cache, evaluate the hit rate of a cache given an address trace, develop a C/C++ cache model and know how to apply address trace analysis optimisations<br>8. Discuss the problems of using caches in a multiprocessor, analyse the operation of several cache coherency protocols and be able to predict the bus traffic given a sequence of CPU "memory" accesses |
| **Module Learning Aims** | This module focuses on the architecture of modern high performance microprocessor systems. Topics covered are basic IA32 and x64 assembly language, procedure calling conventions, the architecture of RISC CPUs, instruction level pipelining, techniques to overcome data, load and control hazards, virtual memory, caches, multiprocessors and cache coherency. This module pays particular attention to issues that improve performance and the close relationship between the hardware and the needs of the software |
| **Module Content** | Topics covered in this module are:<br><br>• Basic IA32 and x64 assembly language<br>• Procedure calling conventions (IA32 and x64)<br>• Mixing C/C++ and assembly language<br>• RISC vs CISC, RISC-1 design criteria and architecture, register windows and delayed jumps<br><br>• Instruction level pipelining, DLX/MIPS pipeline, resolving data, load and control hazards |

| | |
|---|---|
| | • Virtual Memory, memory management units (MMUs), multi-level page tables, TLBs, integration of a MMU into an operating system<br>• Cache organisation (L, K and N), operation, performance, address trace analysis, cache coherency<br>• Multiprocessor architectures, cache coherency protocols (write-through, write-once, Firefly and MESI) |
| **Recommended Reading List** | "Computer Architecture – a Quantitative Approach", Hennessey and Patterson<br>"High Performance Computer Architecture", Harold Stone |
| **Module Pre Requisites** | Assembly language and C/C++ programming |
| **Module Co Requisites** | None |
| **Assessment Details** | Annual assessment<br><br>Exam (2 hrs): 80% typically answer 3 out of 4 questions<br>Coursework + Tutorials: 20%<br><br>Supplemental assessment is by examination ONLY (100%). Students repeating 'off-books' (OBA) are also assessed by examination ONLY (100%) in all examination sessions. |
| **Module approval date** | 3rd August, 2012 |
| **Approved By** | Jeremy Jones |
| **Academic Start Year** | 2012 |
| **Academic Year of Data** | 2018/19 |
| **Website** | https://www.cs.tcd.ie/Jeremy.Jones/CS3021/CS3021.htm |