$ " , . #03) ςςωτ

This course is being offered at Griffith College, CAPA's academic partner in Dublin. The Irish academic system differs from the US, particularly with grading. Griffith College professors expect students to undertake a good deal of independent study to achieve a high mark in their classes. For additional information about this class, please contact the Boston Program Advising Team at 1-800-793-0334.

Computer Hardware

Continuous Assessment: 50%
Exam: 50%

Intended Module Learning Outcomes

On successful completion of this module learners will be able to:

1. Compare and critically analyse approaches to computer architecture.
2. Explain how data encoding and processing is carried out and critically analyse the factors that influence processing performance.
3. Analyse the role and construction of key hardware components and explain how they operate.
4. Explain the role played by storage and analyse and compare approaches to storage.
5. Explain the role played by computer building blocks and peripheral components.
6. Explain how computer networks can be used, created and maintained.
7. Explain the role and outline the functions of an operating system.

Module Objectives

This module aims to support participants as they develop a broadly based, and intellectually challenging framework in the area of computer hardware and low-level software. Participantsl have an awareness of current technologies,

literature and research in the area. Participants are expected to apply the principles of computer hardware to both current and developing technologies. Participants achieve this through developing knowledge and skills in computer hardware. Further, they cultivate an understanding of how the insights and practice from computer hardware technology contribute to the current state of the art in the wider Computer Science landscape.

Module Curriculum

Data encoding and number bases

- Explain how computer hardware encodes data.
- Explain the significance of the number bases used in computing.
- Demonstrate the ability to convert numerical data from one base to another

Digital logic

- Create logic circuits and derive their truth tables and equations.
- Design and simplify logic circuits using Boolean algebra and Karnaugh maps
- Explain the relevance and functioning of sequential logic circuits such as flip-flops.
- Explain the refevance and functioning of shift registers and counters.
- Explain the role of a clock in relation to logic operations.
- Explain how an arithmetic logic unit operates.
- Explain how special purpose logic components such as the half-adder, adder, comparator and decoder function.

Computer Construction, Architectures & Processing

- Von Neumann and Harvard architectures. Processor design approaches
- Techniques for performance enhancement such as caching and parallelism.
- Computer mainboard components such as the processor, buses and bus bridging chips, ports, network and graphics chips.

Storage

- Explain the need for different types of storage.
- Describe computer storage with the help of hierarchical models such as the storage pyramid.
- Distinguish between primary and secondary storage.
- Explain how bit cells can be constructed, written to and accessed.
- Explain how primary storage impacts processing performance.
- Demonstrate the ability to calculate storage effective access time (EAT).
- Explain how secondary storage is implemented and model the construction of typical, commonly available secondary storage devices.
- Demonstrate the ability to calculate capacity, access times and transfer rates for secondary storage devices.
- Explain how redundancy based systems such as RAID operate.

Number Representation

- Explain how integers are represented and stored.
- Distinguish between sign and magnitude representation and two's complement representation.
- Explain how floating point numbers are represented.
- Compare fixed point and floating point number representation approaches.
- Critically analyse a given approach to floating point number representation in relation to parameters such as design approach, range and precision.

Input & Output

- Input and output devices.
- Serial and parallel movement of data.
- Movement of data to and from devices.
- Typical interface types such as serial, parallel, USB, infrared and Bluetooth.
- Interrupts and polling.
- Programmed I/O, I/O instructions, memory-mapped I/O. Direct Memory Access (DMA).

Embedded Systems

- Typical embedded system architectures and applications.
- The place of embedded systems in Computer Science.
- Typical hardware and software.
- Tools.
- Programming and deployment.
- Debugging an embedded system consisting of hardware and software.

Network Hardware

- Creation and configuration of a Local Area Network (LAN) consisting of interfaces, cables and switches.
- Test and diagnosis of network problems using hardware such as cable testers and software tools such as ping, netstat and tracert.