# IT 5305 – Computer Systems II

# (Optional)

## INTRODUCTION

Computer Systems II is offered in the 5th semester and is intended to cover selected concepts of computer architecture and operating systems and the system concepts that support emerging scenarios of cloud and service oriented computing, which are relevant to an IT degree holder

**CREDITS:** 03

## LEARNING OUTCOMES

After successfully completing this course you will be able to:

- Understand the platform dependency of software performance
- Appreciate the emerging paradigms of multicore systems and virtualization that enables cloud computing, in making accurate decisions on applicable software paradigms
- Understand that parallel applications have to be explicitly designed and not merely expected as an outcome of parallelism in hardware
- Acquire basic skills in systems programming

## MINOR MODIFICATIONS

When minor modifications are made to this syllabus, those will be reflected in the Virtual Learning Environment (VLE) and the latest version can be downloaded from the relevant course page of VLE. Please inform your suggestions and comments through the VLE. http://vle.bit.lk

## ONLINE LEARNING MATERIALS AND ACTIVITIES

You can access all learning materials and this syllabus in the VLE: http://vle.bit.lk, if you are a registered student of BIT degree program. It is very important to participate in learning activities given in the VLE to learn this subject.

## FINAL EVALUATION

Final exam of the course will be held at the end of the semester. It is a 2 hour paper of structured questions.

**OUTLINE OF THE SYLLABUS**

| Topics | Hours |
|---|---|
| 1. Uniprocessor architecture | 15 |
| 2. Multiprocessors | 09 |
| 3. OS fundamentals | 15 |
| 4. Virtualisation and service oriented computing | 06 |
| **Total** | **45** |

## REQUIRED MATERIALS

### Main Reading

Learner should use the following user manuals as main reference materials.

- **Ref_1: Computer Organisation and Architecture by William Stalling; 9th Edition; Prentice Hall**

- **Ref_2: Operating Systems Concepts by Silberchatz, Galvin, Gagne; 9th Edition; Wiley**

**DETAILED SYLLABUS**

### 1. Uniprocessor Architecture (15 hrs.)

**Instructional Objectives**

- Express ways to measure the performance of a processor and convert among performance measures
- Express the range and accuracy of a computation that can be handled by a CPU and do basic calculations
- Describe spectrum of CPU paradigms and write simple high level codes and their machine instruction counterparts for each paradigm
- Describe the RISC concept, its expected performance in various domains, instruction set
- Identify whether a RISC architecture is good for a given application domain
- Understand how pipelining, multiple pipelines and threading can enhance CPU performance
- Describe the memory hierarchy of a computer system, the principle on which it works
- Understand the cache concept and cache policies

**Sub Topics**

1.1 Performance metrics
- 1.1.1 CPI, FLOPS, MFLOPS, GFLOPS, TFLOPS and conversion from IPS to CPI
- 1.1.2 CPU Word length effects on range and floating point numerical accuracy

1.2 CPU paradigms
- 1.2.1 Accumulator
- 1.2.2 Register
- 1.2.3 Stack
- 1.2.4 memory/register

1.3 RISC architecture
- 1.3.1 Characteristics: large register file, variable and (not memory) based property, system/network/user environment relative performance
- 1.3.2 ISA: arithmetic/logic, data transfer, flow control and simple examples
- 1.3.3 Performance enhancement principles: instruction pipelines, ILP and TLP; differences and advantages explained through examples

1.4 Memory hierarchy
- 1.4.1 Locality of reference principle
- 1.4.2 Average memory access time expression for a two/three level memory
- 1.4.3 Cache concepts: writing, replacement and multiple copy maintenance policies

## 2. Multiprocessors (09hrs.)

### Instructional Objectives

- Understand the emergence and capabilities of multicore and many core processors
- Explain Amdahl's Law and the need to have inherent parallelism in a task
- Understand how parallelization can be extracted from a simple matrix multiplication example

**Sub Topics**

2.1 Flynn's classification
- 2.1.1 SISD, SIMD, MIMD

2.2 Amdahl's law and extracting parallelism from applications
- 2.2.1 Derivation from first principles
- 2.2.2 Simple matrix multiplication parallelisation

2.3 Shared memory multiprocessors
- 2.3.1 Multicore processors
- 2.3.2 Many core(GPU) processors

## 3. OS fundamentals (15 hrs)

### Instructional Objectives

- Identify main functional components of a multitasking OS
- Define a process and a thread through their contexts
- Understand the workings of a simple multitasking kernel and the concept of job scheduling
- Apply system calls to carry out simple actions
- Understand the concept of virtual memory and how it is implemented
- Explain disk storage management principles and the their relative advantages

### Sub Topics

3.1 Process management
- 3.1.1 Processes and the context of a process
- 3.1.2 Thread and the context of a thread
- 3.1.3 Multitasking kernel with context switching
- 3.1.4 System calls based on POSIX threads for process management
- 3.1.5 Process Scheduling

3.2 Memory management
- 3.2.1 Virtual memory concept
- 3.2.2 Swapping
- 3.2.3 Demand paging

3.3 Storage management
- 3.3.1 File systems concept
- 3.3.2 Directory based structures
- 3.3.3 System calls for file management

## 4. Virtualization and Service oriented computing (06 hrs)

### Instructional Objectives

- Explain the concept of OS virtualization as a resource sharing approach
- Describe a few widely used hypervisors and their effects on application performance
- Understand the concept of the Cloud and virtual on-demand service provision, and the associated risks and practical systems

### Sub Topics

4.1 Introduction to Hypervisors
- 4.1.1 Xen and VMW are: principle, structure and operation in brief

4.2 Introduction to Cloud computing
- 4.2.1 Cloud as a computing service provision
- 4.2.2 Types of clouds: SaaS, IaaS, XaaS
- 4.2.3 Practical systems: Amazon ECC