



## Indicative Content

<b>Topic</b>	<b>Description</b>
<b>Internal microprocessor architecture</b>	Multipurpose registers; Special purpose registers & flags; Segments registers;  Real mode & protected mode addressing;
<b>Addressing Modes</b>	Immediate, Direct, Indirect & Index addressing; Stack addressing;
<b>Assembly programming</b>	Data Movement instructions; Arithmetic & Logic Instructions; Program Control Instructions;
<b>Introduction to Computer Languages</b>	History of programming languages; Compiling & linking; Interpreter based;
<b>Operating System Requirements</b>	Operating system overview & components: memory, process, I/O, file & storage management;  Multitasking requirements: Switching processes and the dispatcher;
<b>Operating system Structures &amp; Services</b>	OS architecture models: Layers, kernel & mini kernel, Hardware abstraction layer;  User & system modes;  System calls: process, file, device, information and communications;
<b>Modular Programming</b>	Assembler & linker; Creating libraries; Using System Calls & Interrupts;
<b>Operating system views</b>	User: Shell; Programmer: API; Engineer: Device drivers;

<b>Virtual Machine</b>	Virtual processors, disks & machines; Implementation & benefits; Java Virtual Machine; API & JDE;
<b>System Bootup</b>	Bootstrapping, BIOS & CMOS settings;
<b>Introduction to DOS operating system</b>	Minimum DOS boot files; DOS Boot sequence; Batch files; Autoexec & config.sys files;
<b>Motherboards &amp; Chip sets</b>	Form factors; Chipset architecture: North/South bridge, Intel hub;

### Teaching and Learning Methods

Students will be taught using a combination of lectures, tutorials and practicals. Practical sessions will be based on lab workbooks. These will contain many small programming assignments to help the students understand the numerous programming constructs being introduced. A number of graded assignments will also be given as part of the course.

### Assessment Methods

Assessment will use both a continuous component and an end of semester examination. The continuous assessment component is used to develop practical skills of programming techniques and will be based both on the lab workbooks and graded assignments / in class tests. Students will be expected to develop efficient, well-documented code, meeting accepted quality standards.

### Primary Reading List

<b>Title</b>	<b>Author</b>	<b>Publisher</b>
The Intel Microprocessors	Barry Brey	PHIPE 2002

### Recommended Reading List

<b>Title</b>	<b>Author</b>	<b>Publisher</b>
The principles of computer hardware	Alan Clements	Oxford
Assembly Language Programming Under OS/2	W.Murray & C.Pappas	McGraw Hill 1988