	THOMAS ADEWUMI UNIVERSITY, OKOCOURSE OUTLINE	
Faculty	**Computing and Applied Sciences**	
Department	Mathematical and Computing Science	
Course Title	**PROGRAMMING IN LOW-LEVEL LANGUAGES**	
Year of Study	2	
Course Code	**CSC 208**	
Credit Hours	2	
Contact Hours	30	
Mode of Delivery	Classroom Lectures, & Laboratory Practical Sessions	
Mode of assessment		Weight%
Continuous Assessment		30%
Final Examination		70%
Total		100%
Course Lecturers and Instructors	Dr. E. K. Olatunji – Lecturer Mr. Taiwo Timothy – Lab Instructor	
Course Description		
Course Objectives	**Some of the following**: 	

Learning Outcomes	Upon completion of this course, students will be able to do the following: • Explain features of the first 3 generations of programming languages • Explain issues that led to the development of Low level languages • Recall the various number systems • Develop algorithms for conversion among the various number systems • Explain underlying principles of program development in a generic assembly language • Explain some important hardware features related to instruction execution: instruction set, Instruction format, Addressing techniques, etc • Explain the registers involved in instruction execution in X86 system • Write simple assembly language program in Microsoft macro Assembler (MASM) □ –	
Teaching and Learning	The class will meet for 3 hours each week.	
Detailed Course Content	Instruction Formats; Addressing Systems ; CPU internal Structure ; Registers, Memory, ALU, CU. Controls - Loops andSwitches; Macro Instructions; Interrupts; File Structures; Storage Control; Relocation; Linking/Loading; Editing; Hardware System Programming using simple machine codes.Problem solving using Assembler/ Machine coding. Intel	

	Instruction Set, Address mode. The relationships between H/L languages and the Computer Architecture that underlies their implementation: basic machine architecture, assembles specification and translation of P/L Block Structured Languages,parameter passing mechanisms.	
Weeks	**Detailed Course Outlines**	**Allocated Time**
Week 1	**1. Introduction to Programming Languages** • Categories of programming language (PL) • Features, advantages and disadvantages of different levels of PL **2. Number systems – Given as assignment to be corrected**	2 hours
Week 2,3	**3. Program development in a Generic Assembly language (AL)** - Structure of a generic AL program - Translation of an AL program -	4 hours
Week 4,5	**4. Review of the Structure of a Computer System** - The peripherals and their functions - CPU – ALU, CU, Memory unit - Registers and their functions in instruction execution - Instruction Execution cycle **5. Machine Instruction** • Instruction set • Instruction & Adress Format, etc	4 hours
Week 6,7	**6. General operand Addressing Techniques** - Indirect, direct, indexed, etc addressing techniques - Advantages and disadvantages of each - Calculation of effective address in each technique - **7. Programming a new machine** **8. Continous assessment I**	4 hours
Week 8, 9	9. Assembly Language for X86 Processor - AL program segment in X86 - AL statement in Microsoft Macro Assembler (MASM) - Some machine and pseudo instruction in MASM	4 hours

	- Sample AL program that displays the "Hello world" -	
Week 10	**10. Program Execution registers in X86 Processor** • The (8) -32bit General purpose registers and their functions • The 16-bit segment registers • Etc □	2 hours
Week 11-14	**11. Fundamentals of MASM** **12. Machine Opcodes in MASM** **13. I/O Operations in X86** **14. Continuous Assessment II** -	8 hours
After week 14	**15. Examination**	

Recommended Reference materials

- 1. Assembly language Programming (An Introductory Text) by E. K. Olatunji
- 2. Computer Science by C.S. French, @ BookPower, 5th edition
- 3. Assembly language for X86 Processors by K. R. Irvine @ 2011, Pearson Education, Inc, Upper Saddle River, New Jersey, USA
- 4. Online Resources