

THOMAS ADEWUMI UNIVERSITY, OKO
COURSE OUTLINE

Faculty	Computing and Applied Sciences	
Department	Mathematical and Computing Science	
Course Title	COMPUTER PROGRAMMING I	
Year of Study	2	
Course Code	CSC 201	
Credit Hours	3	
Contact Hours	45	
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	The course introduces problem-solving methods and algorithm development with an emphasis on good programming techniques and practices for developing good programs.	
Course Objectives	<p>Some of the following:</p> <ul style="list-style-type: none"> • Explanation and demonstration of good programming principles and practices • Algorithm development for simple programming problems in form of pseudo-codes and flowcharts • Fundamentals of Python programming • I/o, Arithmetic and assignment operations in Python 	

	<ul style="list-style-type: none"> • Python List, strings , etc □ 	
Learning Outcomes	<p>Upon completion of this course, students will be able to do the following:</p> <ul style="list-style-type: none"> • Explain what a good program is • Employ good programming principles and practices in program development • Develop algorithms for simple programming problems both with pseudocode and flowcharts • Develop a python program involving standard I/O, arithmetic and assignment operations • Develop simple python program involving selection and iteration control statements • Carry out basic operations on a python list and python string • etc 	
Teaching and Learning	<p>The class will meet for 3 hours each week. There is also a supervised 3 hours per week of Laboratory Practical. Practical will also be given as assignments</p>	
Detailed Course Content	<p>Introduction to problem-solving methods and algorithm development, designing, coding, debugging and documenting programmes using techniques of a good programming language style, programming language and programming algorithm development.</p>	
Weeks	Detailed Course Outlines	Allocated Time
Week 1, 2	<ol style="list-style-type: none"> 1. Introduction to Computer Programming <ul style="list-style-type: none"> - Programs, programming and programming language - Programming process in brief - Sample computer programs - 2. Characteristics of a good program <ul style="list-style-type: none"> - Features of a good program - Programming methodology in brief - 3. Stages in developing good programs 	6 hours

	<ul style="list-style-type: none"> □ • Stages in developing good programs • Principles of good programming methodologies and practices • Types of program errors - 	
Week 3	4. Introduction to Python Programming <ul style="list-style-type: none"> - Intro to python IDLE - Fundamentals of python programming - Sample Python programs - Standard I/O operations - 	3 hours
Week 4,5	5. Algorithm and Pseudocodes <ul style="list-style-type: none"> - Meaning of algorithms and pseudo-codes - Features of an algorithm - Examples of simple algorithms - 6. More Python Programming <ul style="list-style-type: none"> • Review of I/O operations • Assignment and Arithmetic operations • Python Implementation of programs involving only sequence structures 	6 hours
Week 6,7	7. Algorithm & Pseudo-codes of selection Control structure <ul style="list-style-type: none"> - Selection control structure and its flow diagram - Pseudo-code for binary selection control construct 8. Selection Control statements in Python <ul style="list-style-type: none"> - The If-statement - If-Else statement - If-elif statement - 	6 hours
Week 8,9	9. Algorithm & Pseudo-codes of Iteration control structure <ul style="list-style-type: none"> - Iteration control structure and its flow diagram - Pseudo-code for pre-test iteration control construct - 10. Iteration control statements in Python <ul style="list-style-type: none"> - - While statement - For-loop 11. Continuous assessment I	6 hours
Week 10	12. Algorithms and Flowcharts <ul style="list-style-type: none"> • Flowchart symbols • Application of flowchart in solving a programming problem 	3 hours

	-	
Week 11-14	13. Python Strings and operations on them - Kjh - 14. Python Lists and Operations on them - 15. User-defined functions in Python <ul style="list-style-type: none"> • Fundamentals of functions – Built-in and user-defined • Definition of function • Activation of function -	9 hours
After week 14	16. Examination	
Recommended Reference materials <ul style="list-style-type: none"> • Introductory Computer Programming Principles by Olatunji, E.K. • Fundamentals of Python Programming by Richard L. Halterman • A practical Intro to Python Programming by Brian Heinold; Dept of Mathematics and Computer Science, Mount St Mary;s University @ 2012 • Online Resources 		