

	Addis Ababa Science and Technology University				
1	College: Electrical and Mechanical Engineering		Department: Software Engineering		
2	Name of Course:	Fundamentals of Programming I			
	Course Code:	SWEG2103			
3	Synopsis:	<p>Computer programming involves various activities such as understanding and analysis of problems, algorithm design, verification of requirements of algorithms including their correctness and resources consumption, and coding of algorithms. This course is designed to allow students to acquire the three basic skills of programming: problem solving skill (understand how they solve problems), algorithmic skill (knowing how to write the steps of their proposed solutions to specific problems), language skill (learning how to put the solutions in to code and communicate it to the machine).</p> <p>More specifically as an introduction, this course designed to introduce students to an overview of the different concepts of programming and problem solving strategies. Moreover, the learners will be able to identify and demonstrate the basic programming constructors like Variables, Reserved words, Syntax diagram, Constant declarations, Data types, and also the concepts of various programming techniques such as, Conditional statements, Loops, array, strings, pointers (static and dynamic memory management), Error types and error debugging techniques. For each topic, some coding guidelines will be discussed.</p>			
4	Name(s) of Academic Staff:	Prepared by: Eleni Teshome Reviewed by: Biruk Mulatu and Chere Lemma			
5	Semester and Year offered:	Semester:	I	Year:	2
6	Credit Hour:	3			
7	Prerequisite	None			
8	Course Learning Outcome (CLO): At the end of the course, students will be able to:				
	CLO 1	Identify the basic concepts of programming and program development process			

	CLO 2	Demonstrate problem solving skills by analyzing a variety of real-world problems using program solving techniques(flow chart and pseudo code).
	CLO 3	Identify and apply the basic programming constructors' and the various programming techniques such as controlstatements, loops, arrays and pointers in solving problems.
	CLO 4	Develop, debug and test small scale applications

9	Mapping of the course Learning Outcomes to the program Learning Outcomes, Teaching Methods and Assessment																						
	Course Learning Outcomes (CLO)	Program Learning Outcomes (PLO)																					
		PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Teaching Method s				Assessment					
																		Test	Quiz	Final Exam	Project	Lab-report	
		L	T	P	O																		
		CLO1	√												√			√					
		CLO2				√									√		√		√	√	√	√	
		CLO3				√									√		√	√	√	√	√	√	
		CLO4									√						√	√			√	√	√
Indicate the relevancy between the CLO and PLO by ticking “√”on the appropriate relevant box																							
10	Transferable Skills (Skills learned in the course of study which can be useful and utilized in other settings))																						
	1	Develop the habit of time management by organizing, planning, prioritizing their course works as well as on time in class availability and submission of course works (assignment, project work etc.)																					
	2	Able to acquire team spirit and leadership skills through effective management of groups and delegation of various responsibilities, planning and coordination of a variety of tasks related to compute programming																					
	3	Develop communication skills while presenting their programming concepts by articulating and organizing ideas in a logical ways and communicating clearly and concisely both verbally and in written with the course stake holders																					
	4	Acquire problem solving and critical thinking skills																					
11	Distribution of Student Learning Time (SLT)																						
	Course Content Outline										CLO	Teaching and Learning Activities										Total (SLT)	
												Guided learning (F2F) hours				Guided Learning (NF2F)	Independent Learning (NF2F) hours						
												L	T	P	O								

							hours		
Chapter 1: Basic Concepts of programming	1	4					2	4	10
1.1 Basics of Program Development									
o Computer programming and programming language	2								
o Reasons to study programming									
1.2 An overview of Program Development Life Cycle (SDLC)									
1.3 Fundamentals of problem solving techniques (Algorithm design using flow chart and pseudo code)									
1.4 Programming Languages generation									
1.5 Programming Languages Paradigms									

Chapter 2: Basic Programming constructor 2.1 Structure (Anatomy) of a program 2.2 Compilation process 2.3 Preprocessor and Library functions 2.4 Basic Elements, Syntax and Semantics of a high-level programming <ul style="list-style-type: none"> ○ Basic Input/output stream ○ Variables, Data types, reserved words, constants ○ Operators and expression 2.6 Formatted Input-Output 2.7 Programming Errors and Debugging	2	4		6		2	4	16
Chapter 3: Program Flow of Control 3.1 An overview of program flow 3.2 Basics of Logic expression 3.3 Selection Statements <ul style="list-style-type: none"> ○ if selection statements (if, if...else, if...else if...else) ○ Switch statement 3.2 loops <ul style="list-style-type: none"> ○ Loop Basics ○ Loop Statements: for Loop, While loop, do...while loop ○ Other flow controls: continue, break, go to and jump ○ Program termination statements 	2 3 4	8		12		6	10	36
Chapter 4: Arrays and Strings 4.1 Array Basics <ul style="list-style-type: none"> ○ Definition of Arrays ○ Types of Array: One Dimensional Arrays and Multi-dimensional Arrays 	3 4	6		9		4	8	27

<ul style="list-style-type: none"> ○ Declaration and initialization of Arrays ○ Accessing and processing Arrays 								
4.2 Basics of Strings <ul style="list-style-type: none"> ○ Definition of String ○ Declaration and initialization of strings ○ String manipulation and operation <ul style="list-style-type: none"> ▪ String function basics ▪ More on String Functions 								
Chapter 5: Pointers								
5.1 Introduction to memory management	3	4		6		2	6	18
5.2 Pointer definition								
5.3 Pointer implementation								
5.4 Pointers arithmetic and Arrays								
5.5 Dynamic memory allocation								
Total	26		33		16	32	107	
Assessment								

	Continuous Assessment		Percentage Total-50(%)	F 2 F	N F 2 F	S L T
	1	Tests	15%	1	-	1
	2	Quize	5%	½	-	½
	3	Lab-report	15%	2	-	2
	4	Project	15%	3 ½	-	3 ½
	Total					7
	Final Exam		Percentage 50 (%)	F 2 F	N F 2 F	S L T
	Final Exam		5 0 %	3	3	6
	Grand Total SLT					1 2 0
	L = Lecture, T = Tutorial, P = Practical, O = Others, F2F = Face to Face, NF2F = Non Face to Face					
12	Special requirements and resources to deliver the course	1	Software			
		2	Computer Lab			

13	Text books	1	A. B. Chaudhuri (2020), Flowchart and Algorithm Basics: The Art of Programming: The Art of Programming; Mercury Learning and InforMation LLC., 2020
		2	P. Deitel , H. Deitel; C++ how to program, 10th edition, Global Edition (2017)
	References	3	Walter Savitch; Problem Solving With C++ (10th edition), University of California, San Diego, 2018
		4	Diane Zak; An Introduction to Programming with C++ (8th Edition), 2016 Cengage Learning
		5	Gary J. Bronson; C++ For Engineers and Scientists [3rd edition], Course Technology, Cengage Learning, 2010