Subject	Programming in Visual C	**Course Code**	IT112	**Theoretical**	3 hrs / wk
Semester	2	**Prerequisite**	IT111	**Practical**	3 hrs / wk

Program Learning Components

	Topic	Description and Practical Work
Week 1	1. Basic of C programming. 2. Variables and data Type. 3. Operators and Expressions. 4. Basic input and output statement.	Getting Stated With first step for C programming : The General program structure, variables and abstract data type. Arithmetic and logical Expression. Writing First simple C program.
Week 2	**Topic**	**Description and Practical Work**
	1. Variables declaration. 2. Simple built-in Data types. 3. Constant and their use. 4. Memory allocation and binding for variables and constants.	Understanding the sue of variables and constants for data holding during manipulation. Using constants for fixed data values. Getting knowledge how memory space is allocated for variables holding different data types. The role of declaration statement for memory allocation.
Week 3	**Topic**	**Description and Practical Work**
	1. Decision statement and program control flow selection. 2. The simple **IF** Statement. 3. The tow way **IF…ELSE** Statement. 4. The multi selection.	Using the language available selection programming constructs. Conditional selection in different version with one way, tow ways and multiways selections. Intensive examples are to be provided that demonstrate the use and benefits of those constructor.
Week 4	**Topic**	**Description and Practical Work**
	1. Iteration Statements and program compaction. 2. **FOR** Statement as counting loop.	Learning the need for program compaction and eliminating reparations of code parts by building program construct blocks using iteration statements.

	3. **WHILE** statement as pretested loop. 4. **DO...WHILE** statement as post tested loop.	The difference and usage of those statements are clarified by example during lab activities sessions.
Week 5	**Topic**	**Description and Practical Work**
	1. Function and program decomposition. 2. Types of functions. 3. Parameters passing methods between functions. 4. Standard library functions.	Learning how to split programs into functional unit as subprograms. By this splitting the programmers gain tow benefits: avoiding the reparations of code and reusing modules in more than one program as well as using ready developed modules from system library and other programmers.
Week 6	**Topic**	**Description and Practical Work**
	1. Compound data structures. 2. Arrays. 3. Pointers and addresses. 4. Structures and unions. 5. Enumerators.	Collecting related data into one data structures under one name to ease its access, manipulations memory utilization and speeding up the processing time. Practical activities focuses on how to access the elements of each compound data type and what operations can be performed on it.
Week 7	**Topic**	**Description and Practical Work**
	1. String as special arrays of type character. 2. Data inputting and outputting from strings. 3. Handling strings as one unit of data. 4. Library functions for strings.	A string get special attention and treatment in most of modern programming language. The student has to know how VC/VC++ treats string and what library function are available to work with string in word/text processing applications.
Week 8	**Topic**	**Description and Practical Work**
	1. Classes, Objects and inheritance. 2. Overriding, overloading and polymorphism of functions between classes. 3. Interface and abstract	Adding the related functions for related data types/ structures in one programming unit to form a class. This steps enters the students smoothly into VC++ programming. Function organization between classes is treated by over loading

	classes and the general program prototype.	functions in the same class, overriding function between subclasses and polymorphism in different cases in the same hierarchy.
Week 9	**Topic**	**Description and Practical Work**
	1. Mid Term Exam.	Testing the knowledge gained by students so far.
Week 10	**Topic**	**Description and Practical Work**
	1. VC project setting. 2. Massage box Format. 3. Standard controls. 4. Windows messages and notification.	Learning how to start with developing a complete project as an application. Learning the available controls in VC++ to design the graphical user interface GUI.
Week 11	**Topic**	**Description and Practical Work**
	1. Continue with controls. 2. Examples. 3. Command line parsing. 4. Mapping and error handling.	Learning how to start with developing a complete project as an application. Learning the available controls in VC++ to design the graphical user interface GUI.
Week 12	**Topic**	**Description and Practical Work**
	1. Files creation. 2. Data inputting from files. 3. Data outputting to files.	Dealing with files: file type, file formats and file attributes. When using text files and when using Binary file. Importing data from files and exporting data to files. Space management in memory and on disks.
Week 13	**Topic**	**Description and Practical Work**
	1. Introduction to applications development. 2. The capabilities of VC++ for windows Applications. 3. What is next of VC++	Tow lab sessions to develop a semi windows application to highlight the capabilities of VC++ to develop interactive windows application.
Week 14	**Topic**	**Description and Practical Work**
	1. General reviw.	Reviewing what have been studied and what is left for applications development using VC++.

Course Assessment:

Mid Term Exam	Home works and Lap Activities	Final lap Exam	Final Exam
15%	25%	20%	40%

Text Box and References:

1. “Microsoft Visual C++” by Julian T.And Andy Olsen 2002.
2. “Programming in C” 5th Edition by ritch and karnighan.