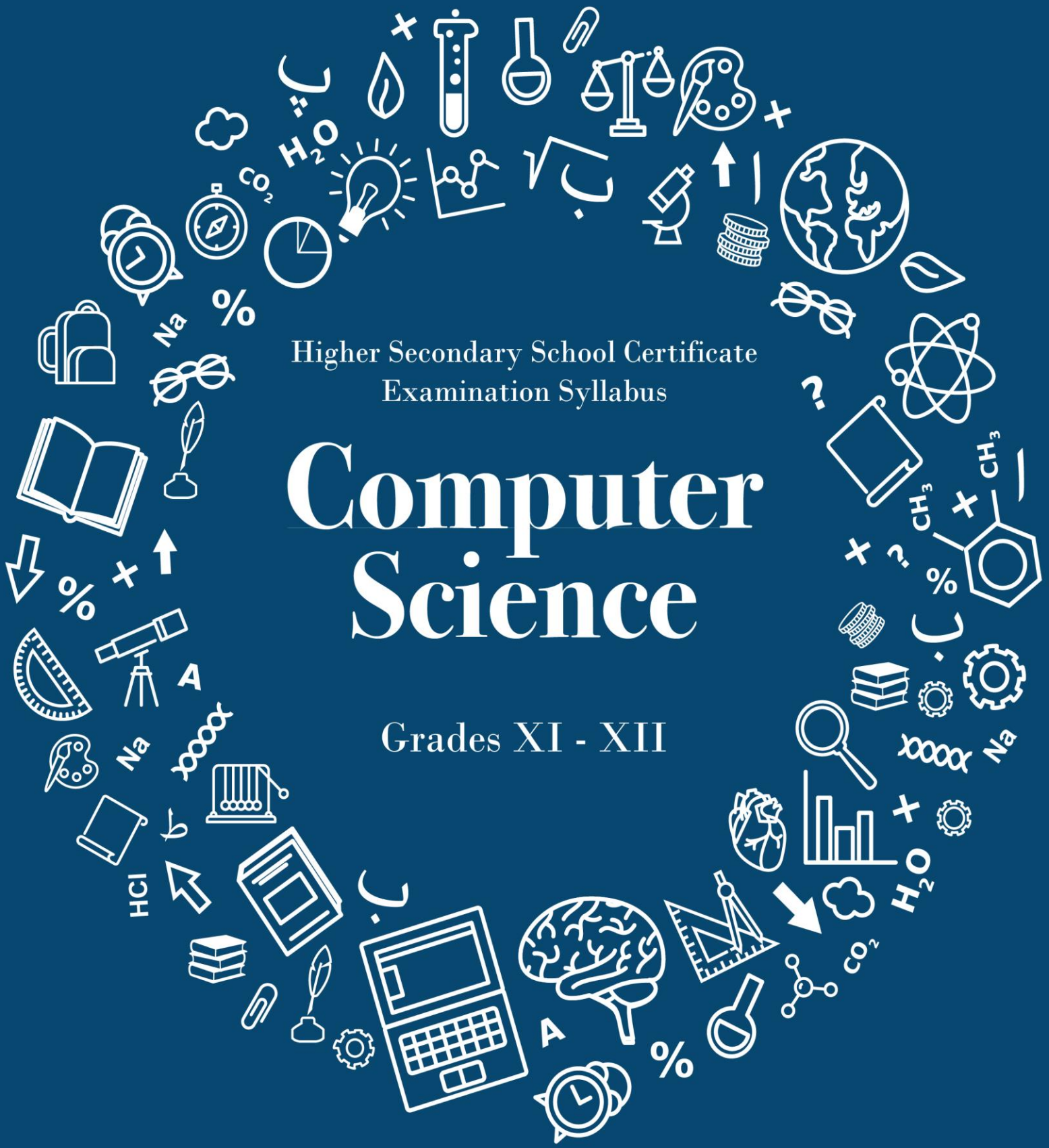
آغا خان یونیورسٹی ایگزامینیشن بورڈ

AGA KHAN UNIVERSITY EXAMINATION BOARD

Higher Secondary School Certificate
Examination Syllabus

Computer Science

Grades XI - XII



Published by
Aga Khan University Examination Board
Block - C, IED - PDC, 1-5/B-VII
Federal B. Area, Karimabad, Karachi, Pakistan.

Latest revision: August 2023
(This syllabus has been periodically revised in 2009 and 2018)

Higher Secondary School Certificate Examination Syllabus

COMPUTER SCIENCE

GRADES XI-XII

This syllabus will be examined in both Annual and Re-sit Examination sessions from Annual Examinations 2024

Table of Contents	Page No.

For queries and feedback

Address: Aga Khan University Examination Board
Block - C, IED - PDC, 1-5/B-VII
Federal B. Area, Karimabad, Karachi, Pakistan.
Phone: (92-21) 3682-7011
Fax: (92-21) 3682-7019
E-mail: examination.board@aku.edu
Website: http://examinationboard.aku.edu
Facebook: www.facebook.com/ akueb

Preface

Established in 2002 through the Pakistan government's ordinance, the Aga Khan University Examination Board (AKU-EB) is country's first private autonomous qualification awarding body for secondary (SSC) and higher secondary (HSSC) school certifications. Its vision is to be a model of excellence and innovation in education in Pakistan and the developing world.

AKU-EB achieves its vision by developing examination syllabi which inculcate conceptual thinking and higher order learning and are aligned with National/ trans-provincial curricula and international standards. AKU-EB revises its syllabi periodically to support the needs of students, teachers and examiners.

The aims of the syllabus review of SSC and HSSC are to:

- Ensure continued compatibility with the goals of the trans-provincial curricula of Pakistan.
- Review the content for inclusion of new knowledge and deletion of obsolete knowledge.
- Review the content for clarity and relevance as per the changing needs of students, teachers and examiners.
- Enhance and strengthen continuation and progression of content both within and across grades IX - XII (SCC and HSSC).
- Ensure the readiness of students for higher education.

During the syllabus review, the needs of all the stakeholders were identified through a needs-assessment survey. Students and teachers of AKU-EB affiliated schools from across Pakistan participated in the survey. Thereafter, a revision panel, which consisted of examiners, teachers of affiliated and non-affiliated schools, teacher trainers and university academicians, reviewed and revised the syllabus following a planned, meticulous and standardised syllabi review process.

The syllabus is organised into topics and subtopics. Each subtopic is further divided into achievable student learning outcomes (SLOs). The SLOs of the cognitive domain are each assigned a cognitive level on which they have to be achieved. These cognitive levels are 'knowledge', 'understanding' and 'application', the latter also including other higher order skills. This is followed by the Exam Specification which gives clear guidance about the weightage of each topic and how the syllabus will be assessed.

The development of the revised syllabus has been made possible by the creativity and relentless hard work of Curriculum and Examination Development unit and the constant support provided by all the other units of AKU-EB. We are particularly thankful to Dr Sohail Qureshi for his very useful feedback on revising the syllabus review process, to Dr Naveed Yousuf for his continued guidance and support throughout the syllabus revision process and to Raabia Hirani for leading the syllabi revision. We are also thankful to all the students and teachers who took part in the needs-assessment survey and to the principals of AKU-EB affiliated schools who made this endeavour possible by facilitating and encouraging their teachers to be a part of the survey and the syllabus revision panel.

With your support and collective hard work, AKU-EB has been able to take the necessary steps to ensure effective implementation of the best international and trans-provincial standards through this syllabus. We are confident that this syllabus will continue to provide the support that is needed by students to progress to the next level of education and we wish the very best to our students and teachers in implementing this syllabus.

Dr Shehzad Jeeva
Chief Executive Officer (CEO), Aga Khan University Examination Board
Associate Professor of Practice, Faculty of Arts and Sciences, Aga Khan University

Understanding of AKU-EB Syllabi

1. The AKU-EB syllabi guide the students, teachers, parents and other stakeholders regarding the topics that will be taught and examined in each grade (IX, X, XI and XII). In each syllabus document, the content progresses from simple to complex, thereby, facilitating a gradual, conceptual learning of the content.

2. The topics of the syllabi are divided into subtopics and **student learning outcomes (SLOs)**. The subtopics and the SLOs define the depth and the breadth at which each topic will be taught, learnt and examined. The syllabi also provide enabling SLOs where needed to scaffold student learning.

3. Each SLO starts with an achievable and assessable **command word** such as describe, relate, evaluate, etc. The purpose of the command words is to direct the attention of teachers and students to specific tasks that the students are expected to undertake in the course of their studies. The examination questions are framed using the same command words or their connotations to elicit evidence of these competencies in students' responses.

4. The topics of the syllabi are grouped into themes derived from the National/ trans-provincial curricula. The connection between various themes and topics is highlighted in the '**concept map**' provided at the beginning of each syllabus. This ensures that students begin to understand the interconnectedness of knowledge, learn conceptually and think critically.

5. The SLOs are classified under three **cognitive levels**: knowledge (K), understanding (U) and application and other higher order skills (A) for effective planning during teaching and learning. Furthermore, it will help to derive multiple choice questions (MCQs), constructed response questions (CRQs) and extended response questions (ERQs) on a rational basis from the subject syllabi.

6. By focusing on the achievement of the SLOs, these syllabi aim to counter the culture of rote memorisation as the preferred method of examination preparation. While suggesting relevant, locally available textbooks for achieving these outcomes, AKU-EB recommends that teachers and students use multiple teaching and learning resources for achieving these outcomes.

7. The syllabi follow a uniform layout for all subjects to make them easier for students and teachers to follow. They act as a bridge between students, teachers and assessment specialists by providing a common framework of student learning outcomes and **exam specifications**.

8. On the whole, the AKU-EB syllabi for Secondary School Certificate (SSC) provide a framework that helps students to acquire conceptual understanding and learn to critically engage with it. This lays a solid foundation for HSSC and beyond.

Subject Rationale of AKU-EB Computer Science

Why study AKU-EB Computer Science?

Computer Science is a field of study that will help you as a student to learn theoretical and practical concepts of computation. You must have used the computer for playing games, surfing the internet, streaming the videos, using media networking websites, preparing presentation and reports, etc. However, this syllabus would add to your knowledge, understanding and skills about some advance uses of computer such as database designing, object-oriented programming (OOP), computer networking, wireless communication, etc.

To be more specific, you will learn the following content areas in the syllabus:

- Computer Hardware
- Computer Software
- Computer Memory and Storage Devices
- Architecture of CPU
- System Unit
- Network Communication and Protocols
- Wireless Communication
- Database Design
- Operating System (OS)
- System Development Life Cycle (SDLC)
- Object-Oriented Programming using C++

AKU-EB HSSC Computer Science syllabus is progression of AKU-EB SSC Computer Science syllabus. Students will learn basics of hardware, software, web designing, computer networks and C programming in SSC and on the basis of this, they will learn advanced concepts of hardware, software, database and object oriented programming in HSSC. AKU-EB HSSC Computer Science syllabus will build the foundation for computer science and other related courses at the university level. It is equally beneficial to those who are interested to pursue their higher education in fields of study other than computer science because of the omnipresence of technology around us.

Where will it take you?

The subject will take you from simple real life problems to different computational solutions and from basic computing technologies to emerging advance hardware and software technologies. The set of skills learned in this syllabus will help you to select your career in the areas of computer science such as:

- Software Development
- Database Administration
- Computer Hardware Engineering
- Computer System Analyst
- Computer Network Engineering
- Web Development

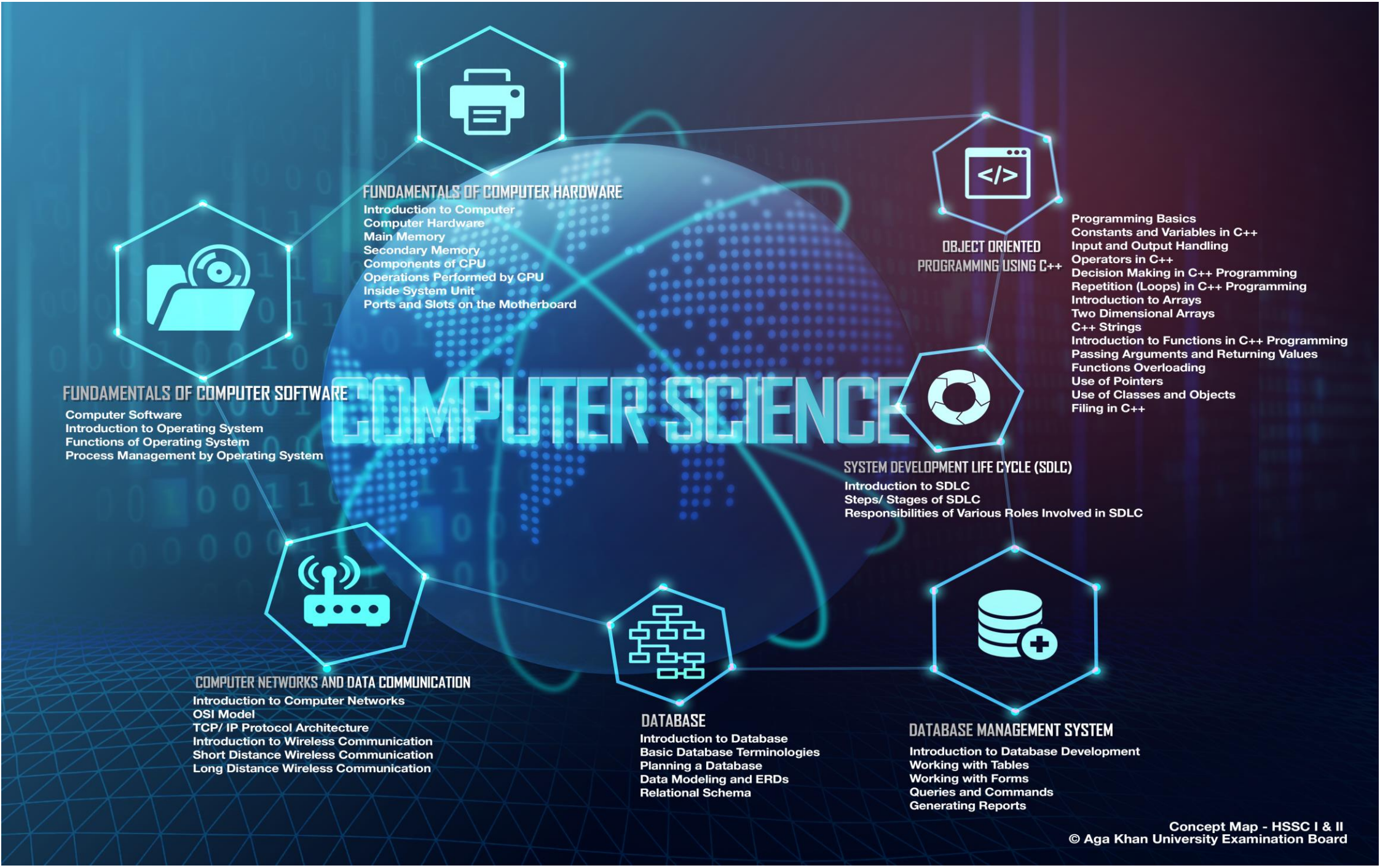
- Web Designing
- Information Security Engineer
- Mobile Applications Development
- Software Quality Assurance
- Content Creation and Blogging
- Graphic Designing

How to approach the syllabus?

The concept map of the syllabus gives an overview of the entire syllabus to teachers and learners. The topics and Student Learning Outcomes (SLOs) guide in details about what has to be achieved through this syllabus. And finally, the Exam Specification guides regarding what will be expected in the examinations.

What is the concept map telling you?

The World Wide Web (WWW) connects the world through a network consisting of hardware and software, using models and protocols to transfer data among them. Similarly, the concepts in AKU-EB syllabus of Computer Science-HSSC are also connected logically and systematically. The weightage of each concept in the syllabus is depicted through the size of the icon in the concept map. Moreover, connections show the suggested sequence of study of the topics. Start exploring and see if you can discover further connections!


COMPUTER SCIENCE
FUNDAMENTALS OF COMPUTER HARDWARE
Introduction to Computer
Computer Hardware
Main Memory
Secondary Memory
Components of CPU
Operations Performed by CPU
Inside System Unit
Ports and Slots on the Motherboard
FUNDAMENTALS OF COMPUTER SOFTWARE
Computer Software
Introduction to Operating System
Functions of Operating System
Process Management by Operating System
OBJECT ORIENTED PROGRAMMING USING C++
Programming Basics
Constants and Variables in C++
Input and Output Handling
Operators in C++
Decision Making in C++ Programming
Repetition (Loops) in C++ Programming
Introduction to Arrays
Two Dimensional Arrays
C++ Strings
Introduction to Functions in C++ Programming
Passing Arguments and Returning Values
Functions Overloading
Use of Pointers
Use of Classes and Objects
Filing in C++
SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)
Introduction to SDLC
Steps/ Stages of SDLC
Responsibilities of Various Roles Involved in SDLC
COMPUTER NETWORKS AND DATA COMMUNICATION
Introduction to Computer Networks
OSI Model
TCP/ IP Protocol Architecture
Introduction to Wireless Communication
Short Distance Wireless Communication
Long Distance Wireless Communication
DATABASE
Introduction to Database
Basic Database Terminologies
Planning a Database
Data Modeling and ERDs
Relational Schema
DATABASE MANAGEMENT SYSTEM
Introduction to Database Development
Working with Tables
Working with Forms
Queries and Commands
Generating Reports
Concept Map - HSSC I & II
© Aga Khan University Examination Board

Student Learning Outcomes of AKU-EB HSSC Computer Science

Part I (Grade XI)

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level[1]		
				K	U	A
1. Basic Concepts of a Computer System		Students should be able to:				
1.1	Introduction to Computer	1.1.1	identify computing devices used for input, process, storage and output;		*	
		1.1.2	describe simple data processing cycle (input, process, output and storage);		*	
		1.1.3	classify computers according to processing speed, size and uses (microcomputer, minicomputer, mainframe computer and supercomputer);		*	
1.2	Computer Software	1.2.1	differentiate among application software, system software and internet applications;		*	
		1.2.2	describe the types of system software i.e. operating system, programming language translators, utility software and device driver;		*	
		1.2.3	describe the use of different application software and their types;		CA[2]	
		1.2.4	compare licensed software, open source software, shareware and freeware with examples;		*	

[1]K = Knowledge, U = Understanding, A = Application and other higher-order cognitive skills
[2]CA = Classroom Activity, not to be assessed under examination conditions

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			K	U	A
	Students should be able to:				
1.3 Computer Hardware	1.3.1	differentiate between manual data entry and automatic data capture devices;		*	
	1.3.2	describe the function of manual data entry devices and their uses in daily life, i.e. keyboard, scanners (2D and 3D), microphone, pointing devices (graphics tablet, joystick, light pen, mouse, touchpad, trackball), digital camera, interactive whiteboard and touch screens (capacitive, resistive and infrared);		*	
	1.3.3	compare the types of automatic data capture devices and their uses in daily life, i.e. magnetic card reader (MCR), magnetic ink character reader (MICR), optical character reader (OCR), barcode reader, optical mark reader (OMR) and quick response (QR) code reader;		*	
	1.3.4	compare the types of output devices and their uses in daily life, i.e. monitor (light emitting diode, liquid crystal display and cathode ray tube), printers (2D and 3D), plotter, projector, computer output microfilm (COM), speakers and head phone.		*	

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			K	U	A
2. Computer Memory and Storage Devices		Students should be able to:			
2.1 Introduction	2.1.1	describe important characteristics of memory devices;		*	
	2.1.2	convert digital storage units from one to another, i.e. bit, nibble, byte, kilobyte (KB), kibibyte (KiB), Megabyte (MB), Mebibyte (MiB), Gigabyte (GB), Gibibyte (GiB), Terabyte (TB), Tebibyte (TiB), Petabyte (PB) and Pebibyte (PiB);			*
2.2 Main Memory	2.2.1	differentiate between primary (main) memory and secondary memory;		*	
	2.2.2	differentiate among Random Access Memory (RAM), Read-Only Memory (ROM) and internal processor memory;		*	
	2.2.3	compare the two types of RAM, i.e. Static RAM (SRAM) and Dynamic RAM (DRAM);		*	
	2.2.4	compare three types of ROM, i.e. Programmable ROM (PROM), Erasable Programmable ROM (EPROM) and Electrically Erasable Programmable ROM (EEPROM);		*	
	2.2.5	describe the role of Basic Input/ Output System (BIOS) in boot-up process;		*	
	2.2.6	compare the three types of cache memory, i.e. Level-1 (L1) cache, Level-2 (L2) cache and Level-3 (L3) cache;		*	
2.3 Secondary Memory	2.3.1	compare magnetic, optical and solid state storage devices;		*	
	2.3.2	describe magnetic storage and its types, i.e. magnetic tapes and disks;		*	
	2.3.3	describe various types of optical storage media, i.e. Compact Discs (CDs), Digital Versatile Discs (DVDs) and Blu-Ray;		*	
	2.3.4	identify CDs and DVDs as CD-ROM, CD-R, CD-RW, DVD-ROM, DVD-R, DVD-RW and DVD-RAM;		*	
	2.3.5	describe solid state storage devices (flash memory and Secure Digital (SD) cards);		*	
	2.3.6	differentiate between sequential and direct access storage.		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
3. Architecture of CPU		Students should be able to:				
3.1	Components of Central Processing Unit (CPU)	3.1.1	describe the role of CPU in computer system;		*	
		3.1.2	describe functions of components of CPU, i.e. Arithmetic and Logic Unit (ALU), Control Unit (CU), registers, cache, internal buses (data bus, address bus and control bus) with the help of a block diagram;		*	
		3.1.3	describe the functions of following registers and their types, i.e. a. general purpose register (Accumulator (AC) and Data Register (DR), Base Register (BR), Counter Register (CR)) b. special purpose register (Instruction Register (IR), Memory Address Register (MAR), Program Counter (PC), Memory Buffer Register (MBR));		*	
3.2	Various Operations Performed by CPU	3.2.1	compare three types of instructions of CPU, i.e. data transfer instructions, data processing instructions and program control instructions;		*	
		3.2.2	compare different types of CPU instruction formats, i.e. zero address instruction, one address instruction and two address instruction;		*	
		3.2.3	describe fetch – decode – execute cycle with the help of a diagram;		*	
		3.2.4	differentiate between Complex Instruction Set Computer (CISC) and Reduced Instruction Set Computer (RISC) architecture;		*	
		3.2.5	differentiate between Intel P4 and advanced micro devices (AMD) Athlon on the basis of clock speed, bus width, cache and architecture.		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
4. System Unit		Students should be able to:				
4.1	Inside System Unit	4.1.1	differentiate between CPU and system unit;		*	
		4.1.2	discuss the role of three major components of system unit, i.e. casing, power supply and motherboard;		CA	
4.2	Ports and Slots on the Motherboard	4.2.1	differentiate among different types of ports, i.e. serial port, parallel port, PS/2 port, Universal Serial Bus (USB) port, fire wire port and High Definition Multimedia Interface (HDMI) port;		*	
		4.2.2	compare the six types of expansion cards, i.e. sound card, video graphics card, modem card, Network Interface Card (NIC), gigabit card and wireless network card;		*	
		4.2.3	differentiate between two categories of memory modules, i.e. Single Inline Memory Module (SIMM) and Dual Inline Memory Module (DIMM).		*	

Topics and Subtopics		Student Learning Outcomes	Cognitive Level		
			K	U	A
5. Network Communication and Protocols		Students should be able to:			
5.1 Introduction to Computer Networks	5.1.1	define a computer network;	*		
	5.1.2	differentiate among different types of networks depending upon coverage area and use: a. nano network b. Body Area Network (BAN) c. Personal Area Network (PAN) d. Near-me-Area Network (NAN) e. Local Area Network (LAN) f. Wide Area Network (WAN) g. Metropolitan Area Network (MAN) h. Internet Area Network (IAN) i. Interplanetary Internet (IPN);		*	
	5.1.3	describe basic network components, i.e. sender, message, medium, protocol, receiver;		*	
	5.1.4	discuss the role of the four data communication devices, i.e. hub, switch, router and gateway;		*	
	5.1.5	differentiate between guided and unguided media and their types, i.e. a. guided media (telephone cable, twisted pair cable, coaxial cable and fibre optic cable) b. unguided media (microwave transmission and satellite communication);		*	
	5.1.6	compare three modes of data transmission, i.e. simplex, half-duplex and full-duplex;		*	
	5.1.7	differentiate between two types of transmission, i.e. serial and parallel;		*	
	5.1.8	differentiate between synchronous and asynchronous transmission of data;		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
		Students should be able to:				
		5.1.9	compare the types of network topologies, (i.e. star, ring, bus, mesh and tree) with the help of diagrams;		*	
5.2	OSI Model	5.2.1	differentiate among seven layers of Open System Interconnection (OSI) model;		*	
		5.2.2	identify protocols and devices used on every layer of OSI model;		*	
5.3	TCP/ IP Protocol Architecture	5.3.1	define Transmission Control Protocol/ Internet Protocol (TCP/ IP) architecture;	*		
		5.3.2	describe function of each layer of TCP/ IP protocol architecture;		*	
		5.3.3	compare the TCP/ IP model with the OSI model;		*	
		5.3.4	differentiate between circuits switching and packet switching with examples;		*	
		5.3.5	differentiate between Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6);		*	
		5.3.6	identify the class of an IPv4 address on the basis of IP range.		*	

Topics and Subtopics		Student Learning Outcomes		Cognitive Level		
				K	U	A
6. Wireless Communication		Students should be able to:				
6.1	Introduction to Wireless Communication	6.1.1	describe wireless networks with examples;		*	
		6.1.2	describe advantages and disadvantages of wireless networks;		*	
		6.1.3	define the following terms: a. radio signals b. radio transceivers c. access point d. line of sight;	*		
		6.1.4	differentiate between long distance and short distance wireless communication;		*	
6.2	Short Distance Wireless Communication	6.2.1	compare the types of short distance wireless technologies, i.e. Wireless Fidelity (Wi-Fi), Worldwide Interoperability for Microwave Access (WiMAX), Bluetooth and Infra-red;		*	
6.3	Long Distance Wireless Communication	6.3.1	explain mobile (cellular) communication;		*	
		6.3.2	describe the role of components required for mobile communication (mobile phone, base station, switching node, landline telephone network);		*	
		6.3.3	describe the limitations of mobile communication systems;		*	
		6.3.4	explain Global Positioning System (GPS);		*	
		6.3.5	differentiate among Geostationary Earth Orbit (GEO) satellites, Medium Earth Orbit (MEO) satellites and Low Earth Orbit (LEO) satellites;		*	
		6.3.6	define the following wireless protocols: a. Wireless Application Protocol (WAP) b. Wired Equivalent Privacy (WEP) protocol.	*		

Topics and Subtopics		Student Learning Outcomes		Cognitive Level		
				K	U	A
7. Database Fundamentals		Students should be able to:				
7.1	Introduction to Database	7.1.1	define database;	*		
		7.1.2	describe advantages of database management system over traditional file management system;		*	
		7.1.3	define the role of Database Administrator (DBA);	*		
		7.1.4	compare the types of database models, i.e. hierarchical database, network database, relational database, object oriented database, object relational database;		*	
		7.1.5	define Structured Query Language (SQL);	*		
		7.1.6	differentiate among the types of SQL languages, i.e. Data Definition Language (DDL), Data Manipulation Language (DML) and Data Control Language (DCL);		*	
7.2	Basic Database Terminologies	7.2.1	define the following terms related to relational database: a. field/ attribute/ column b. record/ tuple/ row c. table d. file e. view f. data type g. key;	*		
		7.2.2	define the data types available in a relational database, i.e. character, integer, real number, Boolean data, date and time;	*		
		7.2.3	differentiate among primary key, candidate key, alternate key, secondary key and foreign key;		*	

Topics and Subtopics		Student Learning Outcomes		Cognitive Level		
				K	U	A
		Students should be able to:				
7.3	Planning a Database	7.3.1	explain the following steps for designing a database: a. problem identification/ definition b. feasibility study c. requirement analysis d. identifying entities and attributes e. assigning names to tables and columns;		*	
7.4	Data Modelling and ERDs	7.4.1	explain the process of data modeling;		*	
		7.4.2	discuss the Entity Relationship Diagram (ERD) and its components, i.e. entity relationship, degree of relationship (cardinality) and attributes;		*	
		7.4.3	show different types of relationships between two entities;			*
		7.4.4	illustrate the difference between cardinality and modality;			*
		7.4.5	draw entity relationship diagram for different scenarios[3];			*
		7.4.6	describe the normalization of relational database.		*	

[3] Refer to Annex B.

Topics and Subtopics		Student Learning Outcomes		Cognitive Level		
				K	U	A
8. Database Development (MS Access 2007 or Above)		Students should be able to:				
8.1	Introduction	8.1.1	describe features of MS-Access database management system;		*	
		8.1.2	define database objects such as table, query, form and report;	*		
		8.1.3	create and save a database;			*
8.2	Working with Tables	8.2.1	create tables in design view and using wizard;			*
		8.2.2	assign[4] appropriate data types to the fields in a table;			*
		8.2.3	set primary key in database table;			*
		8.2.4	create relationship among tables using primary and foreign key;			*
		8.2.5	modify tables and relationships;			*
		8.2.6	sort records in a database table;			*
		8.2.7	add records in a table;			*
		8.2.8	add records in a related table;			*
		8.2.9	modify records in a table;			*
		8.2.10	delete records in a table;			*
		8.2.11	navigate through records in a table;			*
		8.2.12	apply data validation on a field;			*
8.3	Working With Forms	8.3.1	create form using wizard;			*
		8.3.2	create form in design view;			*
		8.3.3	create form for related tables;			*
		8.3.4	identify different form views;		*	
		8.3.5	navigate through records in a form;			*
		8.3.6	add a record using form;			*
		8.3.7	delete a record using form;			*
		8.3.8	modify a record using form;			*

[4] Assign, create, navigate, modify, view, print, set, sort, add, delete and save are operations related to MS Access database.

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
		Students should be able to:				
8.4	Queries in Design View	8.4.1	create query using query wizard;			*
		8.4.2	create SELECT query using query design;			*
		8.4.3	create UPDATE query using query design;			*
		8.4.4	create DELETE query using query design;			*
		8.4.5	create APPEND query using query design;			*
8.5	SQL Queries	8.5.1	write the simple SQL (DML) queries to perform the following: a. insert, delete and update records in table b. select records from table using SELECT statement c. filter records using WHERE statement d. sort records using ORDER BY statement e. group the results using GROUP BY statement f. select records having matching values in both tables using INNER JOIN statement;			*
		8.5.2	write the simple SQL (DDL) queries to perform the following: a. create table b. add primary key c. alter table d. drop table;			*
8.6	Generating Reports	8.6.1	create report using report wizard;			*
		8.6.2	create a report using a query;			*
		8.6.3	describe different layouts of reports;		*	
		8.5.4	view and print reports.			*

Part II (Class XII)

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
9. Operating System (OS)		Students should be able to:				
9.1	Introduction to Operating System	9.1.1	identify the commonly used operating systems;		*	
		9.1.2	list the tasks performed by an operating system;	*		
		9.1.3	differentiate between command line interface and graphical user interface of an operating system;		*	
		9.1.4	differentiate between single user operating system and multi user operating system;		*	
		9.1.5	compare the types of operating system, i.e: a. simple batch system b. multiprogramming batch system c. multitasking operating system d. distributed operating system e. real-time operating system f. parallel processing operating system g. multiprocessor operating system h. embedded operating system i. time-sharing operating system;		*	
9.2	Functions of Operating System	9.2.1	describe the main functions of operating system, i.e. process management, memory management, file management, I/ O system management, secondary storage management, protection system, interrupt handling, network management, command-interpreter;		*	
9.3	Process Management	9.3.1	determine the sequence of execution of processes to get the minimum execution time;			*
		9.3.2	explain the process state diagram including new, running, waiting/ blocked, ready and terminated states of a process;		*	
		9.3.3	differentiate between thread and process;		*	

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			K	U	A
	Students should be able to:				
	9.3.4	differentiate among multithreading, multitasking, multiprogramming and multiprocessing.		*	

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			K	U	A
10. System Development Life Cycle (SDLC)		Students should be able to:			
10.1 System Development Life Cycle (SDLC)	10.1.1	define system;	*		
	10.1.2	describe objectives of SDLC;		*	
	10.1.3	describe the steps/ phases in SDLC;		*	
	10.1.4	describe the responsibilities of the following personnel: a. management team b. system analyst c. project manager d. programmer e. software tester f. customer.		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
11. Introduction to C++ Programming		Students should be able to:				
11.1	Programming Basics	11.1.1	define program, programming language, header files and reserved words;	*		
		11.1.2	explain the basic structure of a basic C++ program including pre-processor directives, main function, body of program;		*	
		11.1.3	define statement terminator and comments in C++ programming;	*		
		11.1.4	use single-line and multiple-line comments in a C++ program;			*
11.2	Constants and Variables in C++ Programming	11.2.1	differentiate among C++ data types;		*	
		11.2.2	differentiate between variable and constant;		*	
		11.2.3	write valid variable names on the basis of variables naming rules;			*
		11.2.4	determine the data types of variables depending upon the values they will store;			*
		11.2.5	write a C++ program in which variables are declared with different data types;			*
		11.2.6	write a program to show the implicit and explicit type casting of variables;			*
		11.2.7	differentiate local and global variables;		*	
11.3	Input Output Handling	11.3.1	write a program to display message and value of variable using **cout** statement;			*
		11.3.2	write a program for taking input during execution of a program using **cin** statement;			*
		11.3.3	write a program that uses getch(), getche(), gets() and puts() functions for I/ O operations;			*

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			K	U	A
	Students should be able to:				
	11.3.4	write a program to print text using the following escape sequences: a. alert – **\a** b. backspace – **\b** c. newline – **\n** d. carriage return – **\r** e. tab – **\t** f. display backslash – \\ g. display single quotation marks – **\'**;			*
	11.3.5	write a program that uses the **setw** and **endl** manipulators;			*
11.4 Operators in C++ Programming	11.4.1	differentiate between operator and operand;		*	
	11.4.2	list the names and symbols of arithmetic operators used in C++ programming;	*		
	11.4.3	write a program using arithmetic operators;			*
	11.4.4	define assignment operator;	*		
	11.4.5	write a simple program using arithmetic assignment operators;			*
	11.4.6	write a program using increment and decrement operators with the postfix and prefix scenario;			*
	11.4.7	list the names and symbols of relational operators;	*		
	11.4.8	write a program using relational operators;			*
	11.4.9	write a program using logical operators;			*
	11.4.10	compare unary, binary and ternary operators;		*	
	11.4.11	write a program to show the order of precedence of arithmetic operators in a C++ program;			*
	11.4.12	define compound expression with reference to C++ programming;	*		
	11.4.13	write a C++ program that uses compound expression.			*

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			K	U	A
12. Control Structures		Students should be able to:			
12.1 Selection Statements in C++ Programming	12.1.1	describe **if**, **if-else** and **else-if** statements;		*	
	12.1.2	write a program for each **if**, **if-else** and **else-if** statements[5];			*
	12.1.3	write a program using **nested if** statement;			*
	12.1.4	explain the use of **switch** statements;		*	
	12.1.5	write a program using **switch** statement;			*
	12.1.6	compare **if**, **if-else**, **else-if** and **switch** statement;		*	
	12.1.7	describe the role of **default** and **break** keywords;			*
	12.1.8	describe the role of **exit** and **return** functions;		*	
	12.1.9	rewrite a program having **if/ if-else/ else-if** statement using **switch** statement;			*
12.2 Repetition (Loop) in C++ Programming	12.2.1	list the types of loops available in C++ programming;	*		
	12.2.2	write a C++ program that uses **for** loop;			*
	12.2.3	write a C++ program that uses **while** loop;			*
	12.2.4	write a C++ program that uses **do while** loop;			*
	12.2.5	differentiate between: a. **for** and **while** loop b. **while** and **do while** loop;		*	
	12.2.6	write a C++ program in which **break** statement is used;			*
	12.2.7	write a C++ program in which **continue** statement is used;			*
	12.2.8	write a C++ program in which **exit( )** function is used;			*
	12.2.9	write a program using **nested for** loop.			*

[5] Refer to Annex C.

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
13. Arrays and Strings		Students should be able to:				
13.1	Introduction to Arrays	13.1.1	define array with reference to programming;	*		
		13.1.2	illustrate the concept of array with respect to array name, size, index number and arrangement of elements in the memory;			*
		13.1.3	write C++ code to declare one dimensional array with different sizes and data types;			*
		13.1.4	write C++ code to initialise one dimensional array;			*
		13.1.5	apply process of traversing using all types of loops for input, output and manipulation of elements;			*
		13.1.6	use **sizeof( )** function to find the size of an array;			*
		13.1.7	write a program which stores numeric values in a one dimensional array using **for** loop and finds the highest, lowest and average values;			*
13.2	Two Dimensional Arrays	13.2.1	explain the concept of two dimensional arrays;		*	
		13.2.2	write C++ code to declare two dimensional array with different sizes and data types;			*
		13.2.3	write C++ code to initialise two dimensional array;			*
		13.2.4	illustrate the process of accessing a particular index number in two dimensional array and writing value on;			*
		13.2.5	write a program for adding/ subtracting/ multiplying two integer matrices of the order up to 4×4;			*
13.3	C++ Strings	13.3.1	define string and the process of declaring string variables;	*		
		13.3.2	explain various methods of initializing a string variable;		*	
		13.3.3	write C++ program to perform various operations on string using string functions , i.e. **strcpy**, **strcat**, **strlen** and **strcmp**.			*

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
14. Functions		Students should be able to:				
14.1	Introduction to Function in C++ Programming	14.1.1	differentiate between predefined and user defined functions;		*	
		14.1.2	describe advantages of user defined functions;		*	
		14.1.3	describe the signature of the functions, i.e. function name, arguments and returning data type;		*	
		14.1.4	explain the process of creating a user defined function, i.e. function declaration, function definition and function call;		*	
		14.1.5	write a program involving a user defined function;			*
		14.1.6	differentiate among the variable types, i.e. local, global, and static;		*	
		14.1.7	differentiate between the formal and actual parameters;		*	
		14.1.8	differentiate between local and global functions;		*	
		14.1.9	define inline function;	*		
14.2	Passing Arguments and Returning Values	14.2.1	write a program to invoke a user defined function and pass arguments by constants, value and reference;			*
		14.2.2	define default argument;	*		
		14.2.3	describe the purpose of return statement;		*	
14.3	Functions Overloading	14.3.1	define function overloading;	*		
		14.3.2	describe advantages of function overloading.		*	

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			K	U	A
15. Pointers		Students should be able to:			
15.1 Use of Pointers	15.1.1	define pointer with respect to C++ programming;	*		
	15.1.2	describe the purpose of reference operator (&);		*	
	15.1.3	describe the purpose of dereference operator(*);		*	
	15.1.4	write C++ code to declare an empty pointer variable of **int**, **double**, **float** and **char** data types;			*
	15.1.5	write a simple program using **&** to return memory address of a variable and storing it in a pointer variable;			*
	15.1.6	write a C++ program that uses pointer variable.			*

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
16. Object Oriented Programming (OOP)		Students should be able to:				
16.1	Classes and Objects in OOP	16.1.1	explain class and object;		*	
		16.1.2	write a C++ program to declare a class along with data members and member functions in its body and create the objects of class in the main( ) function and call member functions of class with the help of objects;			*
16.2	Access Modifiers (Public, Private, Protected and Sealed)	16.2.1	differentiate between private and public access specifiers;		*	
		16.2.2	write a C++ program in which class members are not accessible outside the class (private access specifier);			*
		16.2.3	write a C++ program in which class members are accessible from anywhere where the object is visible (public access specifier);			*
		16.2.4	write a C++ program in which a class uses both public and private access specifiers;			*
16.3	Pillars of OOP (Inheritance, Encapsulation, Abstraction and Polymorphism)	16.3.1	explain the concept of encapsulation;		*	
		16.3.2	differentiate between constructor and destructor;		*	
		16.3.3	differentiate among the types of constructor, i.e. default constructor, user defined constructor and constructor overloading;		*	
		16.3.4	describe inheritance in object oriented programming;		*	
		16.3.5	write the basic syntax of inheritance using base class and derived class;			*
		16.3.6	list the names of three inheritance access specifiers;	*		
		16.3.7	write the syntax of inheritance for three access specifiers;			*
		16.3.8	illustrate types of inheritance in C++ programming using different classes;			*
		16.3.9	describe polymorphism in C++ programming;		*	
		16.3.10	describe the concept of abstraction in C++ programming;		*	
		16.3.11	differentiate between overloading and overriding in OOP.		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				K	U	A
17. File Handling		Students should be able to:				
17.1	File Handling in C++ Programming	17.1.1	differentiate between the binary and the text files in C++ programming;	*		
		17.1.2	compare different modes of opening a file;		*	
		17.1.3	write programs to show the file opening and closing;			*
		17.1.4	write a C++ program to read a file character by character and show output;			*
		17.1.5	write a C++ program to write in a file character by character;			*
		17.1.6	write a C++ program to read a file into strings.			*

Scheme of Assessment

Grade XI

Table 1: Number of Student Learning Outcomes by Cognitive Level

Topic No.	Topic	No. of Sub-Topics	SLOs			Total SLOs
			K	U	A	
1.	Basic Concepts of a Computer System	3	0	11	2	13
2.	Computer Memory and Storage Devices	3	0	13	1	14
3.	Architecture of CPU	2	0	9	0	9
4.	System Unit	2	0	5	0	5
5.	Network Communication and Protocols	3	2	15	0	17
6.	Wireless Communication	3	2	9	0	11
7.	Database Fundamentals	4	5	8	3	16
8.	Database development (MS Access 2007 or above)	6	1	3	30	34
	Total	**26**	**10**	**73**	**36**	**119**
	Percentage		**9**	**61**	**30**	**100**

Table 2: Exam Specification

Topic No.	Topics	Marks Distribution			Total Marks
		MCQs	CRQs	ERQs	
1.	Basic Concepts of a Computer System	10	Total 6 Marks (1 CRQ)		16
2.	Computer Memory and Storage Devices	9	Total 3 Marks (1 CRQ)		12
3.	Architecture of CPU	6	Total 3 Marks (1 CRQ)		9
4.	System Unit	4	Total 3 Marks (1 CRQ)		7
5.	Network Communications and Protocols	5		7 Marks Choose any ONE from TWO	20
6.	Wireless Communications	5	Total 3 Marks (1 CRQ)		
7.	Database Fundamentals	6	Total 3 Marks (1 CRQ)	7 Marks Choose any ONE from TWO	21
8.	Database Development (MS Access 2007 or Above)	5			
	Total	**50**	**21**	**14**	**85**
	Practical*				**15**
	Total				**100**

Grade XII

Table 3: Number of Student Learning Outcomes by Cognitive Level

Topic No.	Topic	No. of Sub-Topics	SLOs			Total SLOs
			K	U	A	
9.	Operating System (OS)	3	1	8	1	10
10.	System Development Life Cycle (SDLC)	1	1	3	0	4
11.	Introduction to C++ Programming	4	6	6	16	28
12.	Control Structures	2	1	6	11	18
13.	Arrays and Strings	3	2	2	11	15
14.	Functions	3	3	9	2	14
15.	Pointers	1	1	2	3	6
16.	Object Oriented Programming (OOP)	3	1	9	7	17
17.	File Handling	1	1	1	4	6
	Total	**21**	**17**	**46**	**55**	**118**
	Percentage		**14**	**39**	**47**	**100**

Table 4: Exam Specification

Topic No.	Topics	Marks Distribution			Total Marks
		MCQs	CRQs	ERQs	
9.	Operating System (OS)	6	Total 3 Marks (1 CRQ)		9
10.	System Development Life Cycle (SDLC)	2	Total 3 Marks (1 CRQ)		5
11.	Introduction to C++ Programming	10	Total 5 Marks (1 CRQ)	7 Marks Choose any ONE from TWO	33
12.	Control Structures	8	Total 3 Marks (1 CRQ)		
13.	Arrays and Strings	3		7 Marks Choose any ONE from TWO	17
14.	Functions	7			
15.	Pointers	3			3
16.	Object Oriented Programming (OOP)	8	Total 4 Marks (1 CRQ)		12
17.	File Handling	3	Total 3 Marks (1 CRQ)		6
	Total	**50**	**21**	**14**	**85**
	Practical*				**15**
	Total				**100**

- Multiple Choice Question (MCQ) requires candidates to choose one best/ correct answer from four options for each question. Each MCQ carries ONE mark.
- Constructed Response Question (CRQ) requires students to respond with a short text (few phrases/ sentences), calculations or diagrams.
- Extended Response Question (ERQ) requires students to answer in a more descriptive form. The answer should be in paragraph form, with diagrams where needed, and address all parts of the question.

- Tables 1 and 3 indicate the number and nature of SLOs in each topic in grades XI and XII. This will serve as a guide in the construction of the examination paper. It also indicates that more emphasis has been given to Understanding (61% in HSSC I and 39% in HSSC II), Application and higher order skills (30% in HSSC I and 47% in HSSC II) to discourage rote memorisation. Tables 1 and 2 however do not translate directly into marks.
- There will be two examinations, one at the end of grade XI and one at the end of grade XII.
- In each grade, the theory paper will be in two parts: paper I and paper II. Both papers will be of duration of 3 hours.
- Paper I theory will consist of 50 compulsory, multiple choice items. These questions will involve four response options.
- Paper II theory will carry 35 marks and consist of a number of compulsory, structured questions and a number of extended response questions. Each extended response question will be presented in an either/or form.
- All constructed response questions will be in a booklet which will also serve as an answer script.

***Practical:**

- In each grade, practical examination will be conducted separate from the theory paper and will consist of 15 marks.
- Practical examination will be based on the list of practical activities given in the examination syllabus. Schools may design their own practical manuals based on these activities.
- Practical journal/ portfolio should be developed by students and endorsed by a figure of authority, such as a teacher or principal, and submitted at the time of the practical examination.
- It is essential for each school to equip its computer laboratories with necessary devices, software etc. to meet the requirements of the practicals in the examination syllabus. Each school will be responsible to make sure that each student is provided the opportunity to do the practicals.

Annex A: List of Acronyms and Their Full Forms

Acronym	Full Form
MCR	Magnetic Card Reader
MICR	Magnetic Ink Character Reader
OCR	Optical Character Reader
OMR	Optical Mark Reader
QR	Quick Response
LED	Light Emitting Diode
LCD	Liquid Crystal Display
CRT	Cathode Ray Tube
COM	Computer Output Microfilm
KB	Kilobyte
MB	Megabyte
GB	Gigabyte
TB	Terabyte
PB	Petabyte
RAM	Random Access Memory
ROM	Read Only Memory
SRAM	Static Random Access Memory
DRAM	Dynamic Random Access Memory
PROM	Programmable Read Only Memory
EPROM	Erasable Programmable Read Only Memory
EEPROM	Electrically Erasable Programmable Read Only Memory
CD	Compact Disc
DVD	Digital Versatile Dics
CD-ROM	Compact Disc Read Only Memory
CD-R	Compact Disc Recordable
CD-RW	Compact Disc Rewriteable
DVD-ROM	Digital Versatile Disc Read Only Memory
DVD-R	Digital Versatile Disc Recordable
DVD-RW	Digital Versatile Disc Rewriteable
DVD-RAM	Digital Versatile Disc Random Access Memory

Acronym	Full Form
CPU	Central Processing Unit
ALU	Arithmetic Logic Unit
CU	Control Unit
ACC	Accumulator
MDR	Memory Data Register
MBR	Memory Buffer Register
IR	Instruction Register
MAR	Memory Address Register
PC	Program Counter
MBR	Memory Buffer Register
CISC	Complex Instruction Set Computing
RISC	Reduced Instruction Set Computing
AMD	Advanced Micro Devices
PS/ 2	Personal System 2
USB	Universal Serial Bus
HDMI	High Definition Multimedia Interface
SIMM	Single In-Line Memory Module
DIMM	Dual In-Line Memory Module
BAN	Body Area Network
PAN	Personal Area Network
NAN	Near-me-Area Network
LAN	Local Area Network
WAN	Wide Area Network
MAN	Metropolitan Area Network
IAN	Internet Area Network
IPN	Interplanetary Internet
OSI	Open System Interconnection
TCP/ IP	Transmission Control Protocol/ Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
Wi-Fi	Wireless Fidelity

Acronym	Full Form
WiMAX	Worldwide Interoperability for Microwave Access
GPS	Global Positioning System
GEO	Geostationary Earth Orbit
MEO	Medium Earth Orbit
LEO	Low Earth Orbit
WAP	Wireless Application Protocol
WEP	Wired Equivalent Privacy
DBA	Database Administrator
SQL	Structured Query Language
DDL	Data Definition Language
DML	Data Manipulation Language
DCL	Data Control Language
ERD	Entity Relationship Diagram
I/ O	Input/ Output
OS	Operating System
SDLC	Software Development Life Cycle

Annex B: Examples of Entity-Relationship Diagram (ERD)[6]

Entity Relationship Diagram
1. Library Management System 2. Student Management System 3. Hotel Management System 4. Hospital Management System 5. Ticket Booking System

Annex C: Examples of C++ Programming[7]

Selection Statement
1. Generate marksheet of students on the basis of inputted marks of different subjects. 2. Show whether a number is positive, negative or zero. 3. Find the maximum and minimum values from inputted numbers. 4. Show whether a number is even or odd. 5. Generate the utility bill on the basis of charges allocated to each unit range. The bill should contain meter number and name of the consumer 6. Identify whether the inputted string is a palindrome or not
Loops
1. Generate a number series (even, odd, prime, Fibonacci etc.) by taking input of starting and ending point. 2. Generate sum of series using loops. 3. Generate table of any inputted number. 4. Calculate factorial of any inputted number. 5. Print pyramid, rectangle and square or any other geometrical shape using nested loops. 6. Input multiple values using loop and calculate average, maximum or minimum value using selection statement. 7. Calculate the number of characters, vowels in an inputted string 8. Reverse an inputted string

[6]Government of Pakistan (2009), Page 34, *National Curriculum for Computer Science IX-XII, Islamabad,* Ministry of Education (Curriculum Wing)
[7] Government of Pakistan (2009), Page 48, *National Curriculum for Computer Science IX-XII, Islamabad,* Ministry of Education (Curriculum Wing)

Functions and Classes
1. Write a user defined function to perform basic arithmetic operations, i.e. add, subtract, multiply and divide. 2. Write a user defined function to calculate area of circle, triangle, parallelogram or any other geometrical shape. 3. Write a user defined function to calculate area, volume of cylinder, sphere, cube and different geometrical shapes. 4. Write a user defined function to calculate average of numbers. 5. Write a user defined function to calculate factorial of a given number. 6. Write C++ programs to define the classes and objects and call them. 7. Write C++ programs to show the concept of inheritance

Annex D: List of Practical

Class XI

S.No	SLO No	Objective	Equipment	Software
		Topic 8: Database Development (MS Access 2007 or Above)		
1.	8.2.1	Creating different tables in design view and wizard.	Computer and Printer	MS Access 2007 or Above
2.	8.2.2	Assign appropriate data types to the fields in a table.		
3.	8.2.6	Sorting records in a table.		
4.	8.2.4	Create relationship among tables using primary and foreign key.		
5.	8.3.1; 8.3.2	Create simple forms using wizards and design view.		
6.	8.3.5	Navigating through records in a form.		
7.	8.3.6; 8.3.7; 8.3.8	Using forms to add, delete or modify a record.		
8.	8.4.1; 8.4.2; 8.4.3; 8.4.4; 8.4.5	Create simple queries (select, update, delete and append) using wizard and design view.		
9.	8.5.1; 8.5.2	Write simple SQL (DDL and DML) queries.		
10.	8.5.1	Create reports using wizard.		
11.	8.5.4	View and print reports.		

Class XII

S.No	SLO No	Objective	Equipment	Software
		Topic 12: Control Structures		
1.	12.1.2	Write a program for each **if**, **if-else** and **else-if**.	Computer	Dev-C ++ or Any Other C Compiler
2.	12.1.3	Write a program using **nested if** statement.		
3.	12.1.5	Write a program using **switch** statement.		
4.	12.2.2	Write a C++ program that uses **for** loop.		
5.	12.2.3	Write a C++ program that uses **while** loop.		
6.	12.2.4	Write a C++ program that uses **do while** loop.		
		Topic 13: Arrays and Strings		
7.	13.1.7	Write a C++ program which stores numeric values in a one dimensional array using **for** loop and finds the highest, lowest and average values.	Computer	Dev-C ++ or Any Other C Compiler
8.	13.2.5	Write a C++ program for adding/ subtracting/ multiplying two integer matrices of the order up to 4x4.		
9.	13.3.3	Write C++ program to perform various operations on string using string fucntions , i.e. **strcpy**, **strcat**, **strlen** and **strcmp**.		

S.No	SLO No	Objective	Equipment	Software
		Topic 14: Functions		
10.	14.1.5	Write a program involving user defined function to perform basic arithmetic operations, i.e. add, subtract, multiply and divide.	Computer	Dev-C ++ or Any Other C Compiler
11.	14.1.6	Write a program involving user defined function to calculate area of circle, triangle and parallelogram.		
12.	14.1.7	Write a program involving use of user defined function to calculate volume of cylinder, sphere and cube.		
13.	14.1.9	Write a program involving user defined function to calculate factorial of a given number.		
14.	14.1.8	Write a program involving user defined function to calculate average of numbers.		
		Topic 15: Pointers		
15.	15.1.5	Write a simple program using **&** to return memory address of a variable and storing it in a pointer variable.	Computer	Dev-C ++ or Any Other C Compiler
16.	15.1.6	Write a C++ program that uses pointer variable.		
		Topic 16: Object Oriented Programming (OOP)		
17.	16.1.2	Write a C++ program to declare a class along with data members and member functions in its body and create the objects of class in the main( ) function and call member functions of class with the help of objects.	Computer	Dev-C ++ or Any Other C Compiler
18.	16.1.6	Write a C++ program in which a class uses both public and private access specifiers.		

Acknowledgements

Aga Khan University Examination Board (AKU-EB) would like to acknowledge the contributions of all those who played an important part in the revision of the AKU-EB HSSC Computer Science syllabus.

We would like to thank **Fahad Ahmad Khan, Former Specialist**, **Computer Science**, AKU-EB, for taking the **subject lead** during the entire process of revising the **Computer Science** syllabus and **Hassan Ud Din, Specialist, Computer Science** for taking the **subject lead** during the development of Exam Specification 2023, Practical Annexures and working as a **post-revision reviewer**.

We are particularly thankful to the **syllabus revision panel** for their time, commitment and effort in revising the syllabus. The panel included:

- **Sobia Zeeshan**
  Bai Virbaijee Soparivala Parsi High School, Karachi
- **Rana Azam Faruqui**
  The Mama Parsi Girls' Secondary School, Karachi
- **Muhammad Azeem**
  Habib Public School, Karachi
- **Muhammad Uwais**
  Aga Khan Higher Secondary School Karimabad, Karachi
- **Inqilab Patel**
  The City School PAF Chapter, Karachi

Furthermore, we thank the following for reviewing the syllabus for **higher education preparedness**, ensuring that the syllabus includes adequate skills and content to effectively prepare students for the next level of education.

- **Dr Zulfiqar Ali Memon**
  Professor, Computer Science, National University of Computer and Emerging Sciences, Karachi

In-house Team

- **Final Reviewer and Advisor: Dr Shehzad Jeeva**
  CEO, AKU-EB

- **Mentor and Guide for Syllabi Review: Dr Naveed Yousuf**
  Former Associate Director, Assessment

- **Syllabi Review Lead: Raabia Hirani**
  Manager, Curriculum Development

- **Syllabi Review Facilitators:**
  Banazeer Yaqoob, Former Associate, Curriculum Development
  Dur Nasab, Associate, Curriculum Development
  Mahrukh Jiwa, Specialist, Middle School Programme

- **Internal Reviewer: Zain-ul-Muluk**
  Manager, Examination Development

- **Learning Resources Reviewer: Ali Bijani**
  Teacher Support and Team

- **Lead Specialist Reviewer: Abid Sohail**
  Lead Specialist, Curriculum & Exam Development

- **Administrative Support:**
  Hanif Shariff, Associate Director, Operations
  Raheel Sadruddin, Manager, Administration, and team

- **Syllabi Feedback Data Analysts:**
  Tooba Farooqui, Former Lead Specialist, Assessment
  Muhammad Kashif, Former Specialist, Assessment
  Muhammad Faheem, Specialist, Assessment

- **Design Support: Hatim Yousuf**
  Specialist, Communications

We would like to thank the students and teachers of affiliated schools from across Pakistan who gave their valuable feedback at various stages of the review process.

We also thank **Shahid Arwani**, Assistant AKU-EB, **Tabassum Ali,** Senior Assistant, AKU-EB and **Anusha Aslam**, Former Assistant, AKU-EB, for data compilation and formatting. In addition, we thank **Noshaba Hameed**, Assistant AKU-EB, and **Naveed Ashiq Ali** for their support during formatting.