آغا خان یونیورسٹی ایگزامینیشن بورڈ

AGA KHAN UNIVERSITY EXAMINATION BOARD

Higher Secondary School Certificate
Examination Syllabus

Computer Science

Grades XI - XII

(Based on New National Curriculum 2022-2023)

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

Latest revision: May 2025
(This syllabus has been periodically revised in 2012, 2017 and 2022)

Higher Secondary School Certificate Examination Syllabus

COMPUTER SCIENCE

GRADES XI-XII

This syllabus will be examined in both Annual and September Examination sessions from Annual Examinations 2026 for Grade XI and Annual Examinations 2027 for Grade XII

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) 3682-7011-8
E-mail: examination.board@aku.edu
Website: https://examinationboard.aku.edu
Facebook: https://www.facebook.com/AKUEBOfficial
Link tree: https://linktr.ee/akuexamboard

Preface

Established in 2002 through the Pakistan government's ordinance, the Aga Khan University Examination Board (AKU-EB) is the country's first private autonomous qualification awarding body for Secondary School Certificate (SSC) and Higher Secondary School Certificate (HSSC). 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 is aligned with the National Curriculum and mapped with provincial curricula and international standards. AKU-EB revises its syllabi periodically to support the needs of students, teachers, and society.

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

- Ensure continued compatibility with the goals of the National Curriculum 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 society.
- Enhance and strengthen continuation and progression of content both within and across grades IX-XII (SSC 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 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 our principal Syllabus Reviewers, Syllabus Revision Panellists, and all other reviewers for their contribution. 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 and students 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 National Curriculum of Pakistan 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 Naveed Yousuf
Chief Executive Officer (CEO), Aga Khan University Examination Board
Associate Professor of Practice, Educational Development, Faculty of Health Science, 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 sub-topics and **student learning outcomes (SLOs)**. The sub-topics 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 during 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 SLOs are classified under the following **cognitive levels** of Bloom's Taxonomy: Remember (R), Understand (U), Apply and beyond [Apply (A), Analyse (An), Evaluate (E), Create (C)]. This is to facilitate effective planning for teaching, learning, and assessment. In addition, some SLOs are identified as Formative Assessments (FA), where applicable.

5. Where applicable, SLOs are followed by **Practical Activities** section to elaborate regarding the assessment in the Practical Examination.

6. The **Examination Specification** is also provided within this syllabus which elucidates the weightage of each topic in the examinations determined on the basis on the content as well as the relevance of the topic.

7. To implement this syllabus, students and teachers can take support from additional material provided by the board to its affiliated schools including **Learning Resource Guides**, **Pacing Guides,** and **Model Papers**.

8. The AKU-EB syllabi for Secondary School Certificate (SSC) and Higher Secondary School Certificate (HSSC) are designed to foster not only conceptual understanding but also critical thinking and problem-solving skills. These syllabi ensure students develop the cognitive, affective and psychomotor skills essential for success at the university and beyond.

Subject Rationale of AKU-EB Computer Science Syllabus

Why study Computer Science?

Studying Computer Science undoubtedly opens doors to an impressive array of career opportunities, ranging from software development to cutting-edge artificial intelligence. This field equips you with essential problem-solving skills, critical thinking, and creativity, enabling you to tackle innovative projects that significantly impact various industries. As technology continues to shape nearly every aspect of modern life, the demand for computer science skills is skyrocketing. The ever-evolving nature of the field guarantees lifelong learning and continuous growth. Whether your passion lies in creating groundbreaking apps, analysing complex data, or building robust systems, computer science offers both intellectual fulfilment and the opportunity to make a tangible difference in the world. Furthermore, the potential for high salaries and job security only enhances its appeal, making it a compelling choice for anyone looking to make their mark in the tech industry.

Why study AKU-EB Computer Science?

Computer Science is a field of study that helps students understand both the theoretical and practical aspects of computation. You may have used a computer for playing games, surfing the internet, streaming videos, using social media, creating presentations and reports. This syllabus, however, will expand your knowledge, understanding and skills by introducing more advanced concepts such as database designing, Object-Oriented Programming (OOP), applications of computers, artificial intelligence, data communication and entrepreneurship.

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

- Computer System
- Cybersecurity
- Computational Thinking
- Database Design
- Data and Analysis
- System Development Life Cycle (SDLC)
- Object-Oriented Programming using Python

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 advanced 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
- System Analyst
- Computer Network Engineering
- Web Development
- Data Science

- Artificial Intelligence
- Machine Learning
- Mobile Applications Development
- Software Quality Assurance
- Content Creation and Blogging
- Graphic Designing

How to approach the syllabus?

The AKU-EB syllabi is carefully designed with a reader-friendly approach to ensure that students and teachers can easily comprehend it, making it functional for teaching, learning and assessment purposes. The syllabus includes following parts:

Subject Rationale	It is an introductory document for students.
Student Learning Outcomes (SLOs)	These guide regarding the details about what has to be achieved.
Exam Specification	It guides regarding what will be expected in the examination.
Practical Activities	These include lab activities to be performed during an academic year.
Additional Resources:	
Pacing Guide	It ensures smooth transition and curricular continuity of a school's academic year. It also predicts the time and pace of syllabi implementation.
Resource Guide	It includes teaching and learning resources for students and teachers.
Model Paper	It guides regarding exam pattern, types of questions and marking scheme.
Command Word Guide	It clarifies expectations regarding the cognitive levels and skills that should be acquired by the students, and which are assessed in its examinations.

Student Learning Outcomes of AKU-EB HSSC Computer Science Syllabus

Part I (Grade XI)

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level[1]		
			R	U	A and beyond
1. Computing Systems	Students should be able to:				
1.1 Data Representation in a Digital Computer	1.1.1	differentiate between analog and digital signals;		*	
	1.1.2	explain the binary data representation using binary pulses, i.e., 0/ low/ off and 1/ high/ on;		*	
1.2 Logic Gates	1.2.1	define the following terms: a. digital logic, b. logic gates and logic circuits, c. truth table;	*		
	1.2.2	explain the following logic gates in terms of the number of inputs and outputs using truth tables: a. AND, b. OR, c. NOT, d. NAND, e. NOR, f. Exclusive OR (XOR), g. Exclusive NOR (XNOR);		*	
	1.2.3	identify logic gates from the truth table;		*	
	1.2.4	explain the uses of logic gates in digital devices;		*	
	1.2.5	represent the logic circuits in the form of a truth table;			A

[1] R = Remember, U = Understand, A = Apply and beyond [Apply (A), Analyse (An), Evaluate (E), Create (C)]

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
	Students should be able to:				
	1.2.6	explain the following Boolean identities: a. Identity Law, b. Distributive Law, c. Associative Law, d. Commutative Law, e. Inverse (Complement) Law, f. De Morgan's Theorem, g. Absorption Law;		*	
	1.2.7	construct a logic circuit for a given real-life problem;			A
1.3 Karnaugh Map (K-Map)	1.3.1	simplify two-variable and three-variable Boolean functions using Karnaugh map;			A
	1.3.2	convert the given algebraic expression into its Sum of Products (SOP) and Product of Sums (POS) forms;			A
1.4 Software Development	1.4.1	explain Software Development Life Cycle (SDLC) and its different phases;		*	
	1.4.2	compare the phases, advantages and disadvantages of the following software development models: a. waterfall model, b. agile model;		*	

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
	Students should be able to:				
1.5 Network Topology	1.5.1	define network topology;	*		
	1.5.2	explain the following network topologies with diagrams: a. bus, b. ring, c. star, d. tree, e. mesh, f. hybrid;		*	
	1.5.3	explain the advantages, disadvantages and applications of network topologies mentioned in SLO # 1.5.2;		*	
	1.5.4	compare scalability and reliability of network topologies;		*	
1.6 Cloud Computing	1.6.1	explain cloud computing, its characteristics, and impact on modern IT infrastructure;		*	
	1.6.2	explain the following cloud computing service models with examples: a. Software as a Service (SaaS), b. Platform as a Service (PaaS), c. Infrastructure as a Service (IaaS);		*	
	1.6.3	explain the following cloud computing types: a. public cloud, b. private cloud, c. community cloud, d. hybrid cloud;		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
1.7	Understanding Cybersecurity Fundamentals	1.7.1	describe cybersecurity and its importance;		*	
		1.7.2	explain the following cybersecurity fundamentals: a. confidentiality, b. integrity, c. availability;		*	
1.8	Common Cybersecurity Threats	1.8.1	explain the following cybersecurity threats and their impact: a. malware, b. phishing, c. pharming, d. Denial of Service (DoS), e. Distributed Denial of Service (DDoS), f. ransomware;		*	
		1.8.2	recommend effective measures to safeguard against cybersecurity threats to enhance digital security;			E
		1.8.3	design a multi-layered security strategy for a given situation that integrates preventive, detective, and corrective measures to mitigate the risks associated with DoS, DDoS, and ransomware attacks;			C
		1.8.4	create a cybersecurity awareness campaign highlighting common threats and effective prevention methods for your community;			FA[2]
1.9	Data Encryption	1.9.1	describe encryption and its importance;		*	
		1.9.2	relate the terms cryptography and encryption;		*	
		1.9.3	compare the following types of encryption techniques in terms of speed, key management, and use cases: a. symmetric, b. asymmetric;		*	

[2] FA= Formative Assessment, not to be assessed under examination conditions

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
2. Computational Thinking and Algorithm	Students should be able to:				
2.1 Computational Thinking and Artefacts	2.1.1 2.1.2 2.1.3	define the term computational thinking; describe different steps of computational thinking; explain the following computational artefacts of software development process: a. computational solution design, b. planning and development of computational artefacts, c. testing computational artefacts;	* 	 * *	
2.2 Algorithm	2.2.1	apply basic computational algorithms to perform the following operations in programming: a. arithmetic, relational, and logical computations, b. conditional (selection) statements, c. iterative (loop-based) processes, d. sorting techniques (insertion sort, bubble sort), e. searching techniques (binary search, linear search).			A

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			R	U	A and beyond
3. Programming Fundamentals (Python)		Students should be able to:			
3.1 Programming Basics	3.1.1	define program and programming language;	*		
	3.1.2	differentiate between program syntax and program semantics;		*	
	3.1.3	explain the following types of programming languages: a. low-level languages: I. machine language, II. assembly language, b. high-level languages: I. procedural language, II. structured language, III. object-oriented language;		*	
	3.1.4	explain the functions of an/a assembler, compiler, and interpreter;		*	
	3.1.5	explain the Python language and its applications;		*	
	3.1.6	explain the Python Integrated Development Environment (IDE);		*	
	3.1.7	describe comments in Python programming;		*	
	3.1.8	write a Python program using single-line and multiple-line comments;			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
3.2	Turtle Graphics	3.2.1 3.2.2 3.2.3	define turtle graphics; use turtle library in Python program; draw different shapes using the following functions: a. **Movement:** get screen( ), right( ) / rt( ), left( )/ lt( ), forward( )/ fd( ), backward( )/ bk( ), b. **Screen Functions:** clear( ), reset( ), stamp( ), clearstamp( ), bgcolor( ), title( ), c. **Changing Turtle Attributes:** shapesize( ), pensize( ), fillcolor( ), pencolor( ), color( ), shape( ), speed( ), begin_fill( )….end_fill(), penup( ), pendown( ), d. **Preset Shapes:** circle( ), dot( );	*		 A A
3.3	Libraries	3.3.1 3.3.2 3.3.3	describe the purpose of libraries; write the syntax of installing a library in Python language; write a Python program that imports the datetime module and uses the following arguments: a. %A, b. %B, c. %C, d. %D, e. %H, f. %I, g. %S, h. %Y;		*	 A A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
3.4	Constants and Variables in Python Programming	3.4.1	define the following basic data types in Python programming: a. char, b. str, c. int, d. float, e. bool;	*		
		3.4.2	differentiate between variable and constant;		*	
		3.4.3	differentiate between local and global variables;		*	
		3.4.4	write valid variable names based on the variable naming rules;			A
		3.4.5	write a Python program to a. utilise different data types, b. demonstrate implicit and explicit type casting of variables;			A
3.5	Input Output Handling	3.5.1	describe the purpose of using eval( ) in Python;		*	
		3.5.2	write a Python program to use the eval( ) function;			A
		3.5.3	write a program to display a message and the value of the variable using the print( ) statement;			A
		3.5.4	write a program for taking input during the execution of a program using the input( ) statement;			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
3.6	Operators in Python Programming	3.6.1	differentiate between operator and operand;		*	
		3.6.2	describe the following types of operators: a. assignment operator, b. membership operators, c. arithmetic operators, d. bitwise operators, e. comparison (relational) operators, f. logical operators;		*	
		3.6.3	write a Python program using the following operators; a. arithmetic operators, b. bitwise operators, c. comparison (relational) operators, d. logical operators;			A
3.7	Selection Statements in Python Programming	3.7.1	describe **if**, **if-else**, and **elif** statements;		*	
		3.7.2	write a Python program using the following statements: a. if, b. if-else, c. if-elif-else, d. nested if, e. pass;			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
3.8	Repetition (Loop) in Python Programming	3.8.1	describe loop and its types in Python programming;		*	
		3.8.2	describe the structure of **for** and **while** loop;		*	
		3.8.3	write a Python program using the following loops: a. for, b. while, c. nested for;			A
		3.8.4	describe the **break**, **continue**, and **exit** statements in Python;		*	
		3.8.5	write a Python program using the following statements: a. break, b. continue, c. exit( ) function;			A
3.9	Debugging in Python	3.9.1	describe the importance of debugging;		*	
		3.9.2	explain different types of bugs (errors) in Python code;		*	
		3.9.3	explain the process of finding bugs using an Integrated Development Environment (IDE);		*	
		3.9.4	explain breakpoints to pause execution;		*	
		3.9.5	apply the process of debugging using the print statement and assert keyword in Python;			A
		3.9.6	apply the process of identifying and resolving bugs using the Python debugger (pdb).			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
4. Data and Analysis		Students should be able to:				
	4.1 Statistical Modelling	4.1.1	define the following terms: a. statistics, b. statistical modeling;	*		
		4.1.2	explain the following statistical modeling techniques (supervised and unsupervised): a. regression, b. classification, c. k-means clustering;		*	
		4.1.3	analyse the process of developing a statistical model in the context of data analysis;			An
		4.1.4	evaluate the effectiveness of statistical modeling in solving real-world problems in the fields of healthcare and finance;			E
	4.2 Experimental Design in Data Science	4.2.1	define the following terms: a. correlation, b. causation, c. population, d. parameters, e. random sample;	*		
		4.2.2	differentiate between observational studies and experimental studies in data science;		*	
		4.2.3	describe the principles of experimental design flow;		*	
		4.2.4	explain the steps involved in the experimental design flow;		*	
	4.3 Statistics and Data Visuals	4.3.1	explain the data collection methods and its types;		*	
		4.3.2	analyse the role of data science in addressing real-world problems, supported by practical examples.			An

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
5. **Applications of Computer Science**	Students should be able to:				
5.1 Introduction to Internet of Things (IoT)	5.1.1	describe Internet of Things (IoT) and its importance in connecting physical devices;		*	
	5.1.2	explain the following components of an IoT system: a. sensors, b. processors, c. connectivity, d. user interface;		*	
	5.1.3	examine IoT's role in improving operations and addressing real-world challenges in manufacturing, agriculture, and healthcare sectors;			An
5.2 Core Technologies Powering IoT	5.2.1	describe key technologies that enable Internet of Things (IoT);		*	
	5.2.2	describe big data analytics (BDA) and its applications in different fields;		*	
	5.2.3	describe communication protocols that facilitate data exchange between Internet of Things (IoT) devices;		*	
	5.2.4	explain embedded systems and their application;		*	
	5.2.5	explain Wireless Sensor Networks (WSN) and their applications in the Internet of Things (IoT);		*	
5.3 Blockchain and Blockchain Networks	5.3.1	define blockchain technology;	*		
	5.3.2	list the technologies that enable blockchain;	*		
	5.3.3	explain blockchain networks and the following types: a. public blockchain network, b. private blockchain network, c. permissioned blockchain network, d. consortium blockchain network;		*	
	5.3.4	explain the integration of blockchain and IoT;		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
5.4	Stakeholders in AI Systems	5.4.1	describe the stakeholders involved in an Artificial Intelligence (AI) system;		*	
		5.4.2	evaluate the benefits and challenges of Artificial Intelligence (AI) in healthcare, education, and industry.			E

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
6. **Impacts of Computing**	Students should be able to:				
6.1 Information Sources	6.1.1	define the following types of information sources: a. books, b. articles, c. websites;	*		
	6.1.2	explain reliable and unreliable sources of information;		*	
	6.1.3	explain the use of search engines and databases to find reliable information;		*	
	6.1.4	discuss the responsible and critical use of social media to access and share information;			E
6.2 Information Privacy and Security	6.2.1	define privacy and security;	*		
	6.2.2	describe the importance of protecting personal information online;		*	
	6.2.3	suggest strategies for safeguarding personal data while accessing information sources;			E
	6.2.4	evaluate the impact of data breaches and the misuse of information on individuals, organisations, and society;			E
6.3 Ethical Use of Information	6.3.1	explain the ethical use of information and intellectual property;		*	
	6.3.2	explain the consequences of misinformation and unethical information practices;		*	
	6.3.3	discuss real-world cases that involve ethical dilemmas related to the use of information and their effects on individuals, organisations, and society;			E

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
	Students should be able to:				
6.4 Connectivity in Computing	6.4.1	define the term 'connectivity' in the context of computing;	*		
	6.4.2	describe the following components involved in achieving connectivity: a. devices, b. networks, c. protocols, d. internet;		*	
	6.4.3	describe the following communication technologies enabled by computing: a. social media platforms, b. email, c. instant messaging, d. video conferencing;		*	
6.5 Environmental Impacts of Connectivity in Computing	6.5.1	explain the concept of green computing;		*	
	6.5.2	analyse the positive and negative environmental impacts of connectivity in computing across fields such as cloud computing and smart devices;			An
	6.5.3	examine the environmental impact of data centres, considering factors such as energy consumption and e-waste;			An
6.6 Cultural Impacts of Connectivity	6.6.1	analyse the role of enhanced connectivity in facilitating global cultural exchange;			An
	6.6.2	discuss the preservation or loss of local cultures in the digital age;			E
6.7 Human Impact of Connectivity	6.7.1	discuss the positive and negative impacts of digital connectivity on human behaviour and society, including work and employment.			E

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
7. Digital Literacy	Students should be able to:				
7.1 Introduction to Digital Literacy	7.1.1	describe digital literacy and its importance;		*	
	7.1.2	explain the role of digital literacy in academic and everyday life;		*	
7.2 Introduction to Advanced Search Techniques	7.2.1	describe advanced search techniques and their importance;		*	
	7.2.2	differentiate between basic and advanced searches;		*	
7.3 Designing Data-Collection Approaches	7.3.1	describe the following data collection methods: a. qualitative, b. quantitative;		*	
	7.3.2	discuss the use of prototypes and simulations in data collection, including their purpose and advantages;			E
	7.3.3	design a data-collection plan using qualitative interviews by: a. formulating research questions, b. developing interview guides and protocols;			C
	7.3.4	create surveys for quantitative data collection by: a. writing clear and unbiased survey questions, b. selecting appropriate survey platforms and tools.			C

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
8. Entrepreneurship in the Digital Age	Students should be able to:				
8.1 Designing and Iterating User Centered Prototypes	8.1.1	apply basic design principles to develop a low-fidelity prototype that addresses a specific user problem;			FA
	8.1.2	illustrate the user journey and key features of the prototype using visual tools such as sketches or storyboards;			FA
	8.1.3	determine improvements in the prototype based on user testing and feedback;			FA
	8.1.4	analyse test results to identify whether the prototype meets user needs or validates a business assumption;			FA
	8.1.5	evaluate the effectiveness of the final prototype in addressing user needs and suggest future improvements.			FA

Student Learning Outcomes of AKU-EB HSSC Computer Science Syllabus

Grade XII

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level[1]		
				R	U	A and beyond
9. Computing Systems		Students should be able to:				
9.1	Usability, Security and Accessibility of Devices	9.1.1	describe the use of digital device in terms of ease of use and efficiency;		*	
		9.1.2	describe the security features of devices and best practices for data protection;		*	
		9.1.3	explain the ways in which accessibility features in devices support users with disabilities;		*	
9.2	Human Interaction with Computer Systems in Terms of Usability	9.2.1	describe Human-Computer Interaction (HCI) and its importance;		*	
		9.2.2	analyse the influence of Human-Computer Interaction (HCI) on user efficiency and task effectiveness;			An
9.3	Design Consideration	9.3.1	describe HCI design considerations and their effects on usability and system performance;		*	
		9.3.2	analyse the trade-offs involved in designing HCI in the context of: a. security, b. user-friendliness, c. cost-effectiveness.			An

[1]R = Remember, U = Understand, A = Apply and beyond [Apply (A), Analyse (An), Evaluate (E), Create (C)]

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
10. Computational Thinking and Algorithm						
10.1	Complex Data Structure	10.1.1	define data structure;	*		
		10.1.2	describe the following types of data structures: a. list, b. array, c. tree, d. stack, e. queue;		*	
10.2	Operations on Data Structures	10.2.1	describe the traversal operation on an array;		*	
		10.2.2	describe the enqueue and dequeue operations in a queue;		*	
		10.2.3	explain the push and pop operations in a stack;		*	
		10.2.4	explain the following operations on a tree: a. traversing, I. in order, II. pre-order, III. post order, b. binary search;		*	
10.3	Applications of Data Structures	10.3.1	explain the application of tree data structures;		*	
		10.3.2	explain the application of the binary search algorithms;		*	
		10.3.3	apply tree data structures to represent hierarchical data models such as file systems or organisational charts;			A
		10.3.4	use the binary search algorithm to locate elements in sorted data efficiently;			A
		10.3.5	illustrate the use of tree traversal techniques in data retrieval and manipulation tasks.			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
11. **Programming Fundamentals (Python)**		Students should be able to:				
11.1	Introduction to Function in Python Programming	11.1.1	describe predefined and user-defined functions;		*	
		11.1.2	describe the advantages of user-defined functions;		*	
		11.1.3	describe the signature of the functions, i.e., function name, arguments, and return value;		*	
		11.1.4	explain the process of creating a user-defined function, i.e., function declaration, function definition, and function call;		*	
		11.1.5	differentiate among the following variable types used in functions: a. local, b. global;		*	
		11.1.6	write a program involving a user-defined function;			A
11.2	Passing Arguments and Returning Values	11.2.1	define default argument;	*		
		11.2.2	describe the purpose of the return statement;		*	
		11.2.3	write a program that demonstrates the return statement;			A
		11.2.4	write a program to invoke a user-defined function and pass arguments by constant, value, and reference;			A
11.3	Classes and Objects in Object-Oriented Programming (OOP)	11.3.1	describe Object-Oriented Programming (OOP) and functional programming;		*	
		11.3.2	compare the advantages and disadvantages of Object-Oriented Programming (OOP) and functional programming;		*	
		11.3.3	explain class and object in terms of OOP;		*	
		11.3.4	write a Python program to a. declare a class with data members and member functions defined within its body, b. create objects of the class in the main() function and call its member functions using those objects;			A

Topics and Sub-topics	Student Learning Outcomes		Cognitive Level		
			R	U	A and beyond
	Students should be able to:				
11.4 Access Modifiers (Public, Private and Protected)	11.4.1	differentiate between private and public access specifiers in Object-Oriented Programming (OOP);		*	
	11.4.2	write a Python program that demonstrates private access to class members using name mangling;			A
	11.4.3	write a Python program that demonstrates public access to class members;			A
	11.4.4	write a Python program that defines a class with both public and private members and demonstrates their usage;			A
11.5 Pillars of OOP (Inheritance, Encapsulation, Abstraction and Polymorphism)	11.5.1	describe inheritance and its access specifiers in Python programming;		*	
	11.5.2	explain the types of inheritance in Python programming using different classes;		*	
	11.5.3	explain encapsulation in Python programming;		*	
	11.5.4	explain polymorphism in Python programming;		*	
	11.5.5	explain the function of constructors and destructors in Object-Oriented Programming (OOP);		*	
	11.5.6	differentiate among the types of constructors, i.e., default constructor, user-defined constructor, and constructor overloading;		*	
	11.5.7	differentiate between overloading and overriding in OOP;		*	
	11.5.8	write a program of inheritance using the base class and the derived class;			A
	11.5.9	write a program of inheritance for three access specifiers;			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
11.6	Working with List	11.6.1	explain the concept of a list in Python programming including its name, size, index number, and arrangement of elements in the memory;		*	
		11.6.2	explain the purpose of storing elements in an ordered list;		*	
		11.6.3	write a Python program to use a one-dimensional list with different sizes and data types;			A
		11.6.4	write a Python program to create, add, modify, and remove items in a list;			A
		11.6.5	write a Python program that stores numeric values in a one-dimensional list using a for loop and find the highest, lowest, average, and search for a value;			A
11.7	Working with Dictionaries	11.7.1	describe dictionary and its uses in Python programming;		*	
		11.7.2	write a Python program to construct and retrieve values from a dictionary in Python programming;			A
		11.7.3	write a Python program to find a value in a dictionary using the key value;			A
11.8	File Handling in Python Programming	11.8.1	describe the importance of disk I/O;		*	
		11.8.2	explain file handling methods and the following file operations in Python programming: a. create, b. read, c. write, d. append;		*	
		11.8.3	write a Python program that performs the following file operations: a. create and write data to a file, b. read from and append data to the file, c. write new data and then read it from the file;			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
11.9	Introduction to Database (MS SQL)	11.9.1	describe database and its importance;		*	
		11.9.2	differentiate between Database Management System (DBMS) and Relational Database Management System (RDBMS);		*	
		11.9.3	describe the MS SQL database;		*	
		11.9.4	describe the following keys: a. primary key, b. foreign key, c. candidate key, d. alternate key e. secondary key;		*	
		11.9.5	construct a database and tables using MS SQL;			A
11.10	Data Normalisation	11.10.1	describe data normalisation and its importance in a database;		*	
		11.10.2	explain 1NF, 2NF and 3NF forms of data normalisation;		*	
11.11	Structure Query Language (SQL)	11.11.1	describe the Structured Query Language (SQL) and its importance;		*	
		11.11.2	explain the following types of SQL statements: a. Data Manipulation Language (DML), b. Data Definition Language (DDL), c. Data Control Language (DCL), d. Transaction Control Language (TCL), e. Data Query Language (DQL);		*	
		11.11.3	explain data manipulation language (DML) statements ;		*	
		11.11.4	apply SELECT statement with clauses: a. WHERE, b. Order by;			A
11.12	Connecting Database with Python	11.12.1	explain the process of connecting a database with Python;		*	
		11.12.2	write a program to connect a database with Python.			A

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
12. Data and Analysis		Students should be able to:				
12.1	Introduction to Machine Learning (ML)	12.1.1	define Machine Learning (ML);	*		
		12.1.2	differentiate between Machine Learning (ML) and rule-based algorithm;		*	
		12.1.3	explain the following basics of Machine Learning (ML) model building: a. feature engineering, b. train test split, c. model assessment;		*	
		12.1.4	explain the following Machine Learning (ML) algorithms: a. regression, b. clustering, c. classification;		*	
12.2	Hypothesis and Hypothesis Testing	12.2.1	describe hypothesis and its role in statistical analysis;		*	
		12.2.2	describe the null hypothesis and the alternative hypothesis;		*	
		12.2.3	explain P value and significance testing;		*	
		12.2.4	explain the following tests: a. f-test, b. chi-square test, c. ANOVA;		*	
12.3	Key Model Performance Metrics	12.3.1	explain the following performance evaluation techniques; a. accuracy, b. precision, c. recall, d. f1 score, e. rand index, f. ROC (Receiver Operating Characteristic), g. RMSE (Root Mean Square Error).		*	

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			R	U	A and beyond
13. Applications of Computer Science		Students should be able to:			
13.1 Applications of IoT (Internet of Things), Cloud Computing and Blockchain in Pakistan	13.1.1	describe the following key applications of IoT in Pakistan: a. smart homes, b. agriculture, c. smart cities, d. healthcare monitoring systems;		*	
	13.1.2	explain the importance of cloud computing in Pakistan for data storage;		*	
	13.1.3	evaluate the benefits and challenges of cloud computing for data management in Pakistan;			E
13.2 Deep Learning and its Application	13.2.1	explain the relationship between deep learning networks and neural networks;		*	
	13.2.2	analyse the applications of neural networks, including deep learning models, in fields such as healthcare, finance, and autonomous vehicles.			An

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
14. Impacts of Computing		Students should be able to:				
14.1	Introduction to Privacy and Security	14.1.1	define the terms, privacy and security;	*		
		14.1.2	describe the following common types of cyberattacks: a. Denial of Service (DoS) and Distributed Denial of Service (DDoS) attack, b. ransomware, c. spyware, d. viruses, e. phishing, f. Domain Name System (DNS) spoofing, g. waterhole attack, h. fake Wireless Access Point (WAP), i. eavesdropping;		*	
		14.1.3	explain the following basic security methods: a. strong passwords, b. Two-Factor Authentication (2FA), c. antivirus and anti-malware software, d. checking logs, e. firewall, f. Intrusion Detection System (IDS); g. encryption;		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
		Students should be able to:				
14.2	Security Protocols	14.2.1	explain the following network security protocols: a. Secure Socket Layer (SSL) b. Transport Layer Security (TLS), c. Internet Protocol Security (IPsec), d. Secure Shell (SSH), e. Hypertext Transfer Protocol (HTTP), f. Hypertext Transfer Protocol Secure (HTTPS), g. Wireless Protected Access (WPA/ WPA2/ WPA3), h. File Transfer Protocol Secure (FTPS), i. Simple Network Management Protocol (SNMP);		*	
		14.2.2	differentiate between HTTP and HTTPS;		*	
14.3	Cryptography	14.3.1	define cryptography;	*		
		14.3.2	differentiate between cybersecurity and cryptography;		*	
		14.3.3	explain the following different types of ciphers: a. plain, b. ceaser, c. shift, d. substitution, e. transposition, f. stream cipher, g. block cipher.		*	

Topics and Sub-topics		Student Learning Outcomes		Cognitive Level		
				R	U	A and beyond
15.	**Digital Literacy**	Students should be able to:				
15.1	Understanding Advance Searches and Digital Tools in Research	15.1.1 15.1.2 15.1.3 15.1.4	define research questions/ problems; describe a case study in research; analyse the effectiveness of digital tools in communicating research results and conclusions; evaluate the effectiveness of advanced search techniques in locating information on a research topic;	*	*	An E
15.2	Plan, Execute, and Communicate Research	15.2.1	design a data collection plan to answer a research question and present the data using appropriate digital tools.			FA[2]

[2]FA= Formative Assessment, not to be assessed under examination conditions

Topics and Sub-topics		Student Learning Outcomes	Cognitive Level		
			R	U	A and beyond
16. Entrepreneurship in the Digital Age		Students should be able to:			
16.1 Understanding Minimum Viable Product (MVP) and Business Ideas	16.1.1	define Minimum Viable Product (MVP);	*		
	16.1.2	explain the key characteristics of a successful MVP;		*	
	16.1.3	differentiate between MVP and prototype;		*	
	16.1.4	explain the riskiest assumption (Business Model);		*	
	16.1.5	apply the principles of MVP development to design a simple prototype for a specific user need;			A
	16.1.6	analyse the process of designing a test using an MVP;			An
	16.1.7	analyse case studies of successful businesses that created their MVPs.			An

Practical Activities of AKU-EB HSSC Computer Science Syllabus

Student Learning Outcomes

Content Covered	Actions Performed	Attitude Developed
• Introduction to Hypertext Markup Language (HTML) • Designing Webpage: Text Formatting • Designing Webpage: Creating List • Designing Webpage: Images and Background • Programming Basics (JavaScript) • Operators in JavaScript • I/O Functions in JavaScript • Conditional (Selection) Control Structure • Loop Structure	• Follow the procedure provided in the Lab Guidelines. • Start the computer to perform the task. • Perform the task with the help of the given method. • Execute simple programs using a selected programming language. • Demonstrate cybersecurity principles to ensure safe data handling, secure authentication, and protection from malware.	• Ensure safety of yourself, others around you, and your surroundings. • Demonstrate a computational mindset by asking questions. • Display ethical dealings and practices while performing task. • Show willingness to solve problems and challenges. • Show self-reliance and cooperation when working independently and in a group setting respectively.

Topic-Wise Practical Activities

Part I (Grade XI)

S. No	Practical Activity	Equipment	Software
	Topic 3: Programming Fundamentals (Python)		
1.	Write a Python program to use single-line and multiple-line comments.	Computer	Python IDE
2.	Write a Python program to use different data types.		
3.	Write a Python program to demonstrate the eval( ) function.		
4.	Write a Python program to display a message and the value of the variable using the print( ) statement.		
5.	Write a Python program for taking input during the execution of a program using the input( ) statement.		
6.	Write a Python program using arithmetic operators.		
7.	Write a Python program using bitwise operators.		
8.	Write a Python program using comparison (relational) operators.		
9.	Write a Python program using logical operators.		
10.	Write a Python program for each if, if-else, and if-elif-else statement.		
11.	Write a Python program using a nested if statement.		

12.	Write a Python program using the pass statement.	Computer	Python IDE
13.	Write a Python program that uses a for loop.		
14.	Write a Python program that uses a while loop.		
15.	Write a Python program that uses break statement.		
16.	Write a Python program that uses the continue statement.		
17.	Write a Python program that uses exit( ) function.		
18.	Write a Python program using a nested for loop.		

Part II (Grade XII)

S. No	Practical Activity	Equipment	Software
	Topic 10: Programming Fundamentals (Python)		
1.	Write a Python program involving a user-defined function.	Computer	Python IDE
2.	Write a Python program to invoke a user-defined function and pass arguments by constant, value, and reference.		
3.	Write a Python program to demonstrate the return statement.		
4.	Write a Python program that defines a class with data members and member functions. In the main( ) function, create objects of the class and use them to call its member functions.		
5.	Write a Python program in which class members are accessible wherever the object is visible (public access specifier).		
6.	Write the basic syntax of inheritance using the base class and the derived class.		
7.	Write a Python program to create, add, modify, and remove items in a list.		
8.	Write a Python program that stores numeric values in a one-dimensional list using a for loop and find the highest, lowest, average, and search for a value.		
9.	Write a Python program to find a value in a list.		

10.	Write a Python program to construct and retrieve values from a dictionary.	Computer	Python IDE
11.	Write a Python program to find a value in a dictionary using the key value.		
12.	Write a Python program that creates and writes data to a file.		
13.	Write a Python program that reads and appends data to the file .		
14.	Write a Python program that writes and reads data to the file program involving user defined function to calculate average of numbers.		
	Topic 10: Introduction to Database and SQL		
15.	Construct a database and tables using MS SQL .	Computer	MS SQL
16.	Apply SELECT statement with clauses: a. WHERE, b. Order by.		

Annexure A: List of Acronyms and Their Full Forms

Acronym	Full Form
NAND	NOT AND
NOR	NOT OR
XOR	Exclusive OR
XNOR	Exclusive NOT OR
POS	Product of sum
SOP	Sum of Product
SDLC	System Development Life Cycle
SaaS	Software as a Service
PaaS	Platform as a Service
IaaS	Infrastructure as a Service
DoS	Denial of Service
DDoS	Distributed Denial of Service
IDE	Integrated Development Environment
Pdb	Python Debugger
IoT	Internet of things
WSN	Wireless Sensor Network
BDA	Big Data Analytics
AI	Artificial Intelligence
HCI	Human Computer Interaction
OOP	Object Oriented Programming
DBMS	Database Management System
RDBMS	Relational Database Management System
SQL	Structured Query Language
1NF	First Normal Form
2NF	Second Normal Form
3NF	Third Normal Form
DML	Data Manipulation Language
DDL	Data Definition Language
DCL	Data Control Language
TCL	Transaction Control Language

Acronym	Full Form
DQL	Data Query Language
ML	Machine Learning
ANOVA	Analysis of Variance
ROC	Receiver Operating Characteristic
RMSE	Root Mean Square Error
WAP	Wireless Access Point
2FA	Two-Factor Authentication
MFA	Multifactor Authentication
IDS	Intrusion Detection System
SSL	Secure Sockets Layer
TSL	Transport Layer Security
IPsec	Internet Protocol Security
SSH	Secure Shell
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
FTPS	File Transfer Protocol Secure
SNMP	Simple Network Management Protocol
MVP	Minimum Viable Product
TCP/ IP	Transmission Control Protocol/ Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
Wi-Fi	Wireless Fidelity

Annexure B: Examples of Python Programming

Selection Statement
1. Generate a marksheet of students based on 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 based on charges allocated to each unit range. The bill should contain the meter number and the name of the consumer. 6. Identify the palindrome in the inputted string.
Loops
1. Generate a number series (even, odd, prime, Fibonacci, etc.) by taking the starting and ending point input. 2. Generate the sum of the series using loops. 3. Generate a table of any inputted number. 4. Calculate the factorial of any inputted number. 5. Print pyramid, rectangle, square, or any other geometrical shape using nestedloops. 6. Input multiple values using a loop and calculate the average, maximum, or minimum value using a selection statement. 7. Calculate the number of characters and vowels in an inputted string. 8. Reverse an inputted string.
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 the area of a circle, triangle, parallelogram or any other geometrical shape. 3. Write a user defined function to calculate area and volume of cylinder, sphere, cube, and different geometrical shapes. 4. Write a user-defined function to calculate the average of numbers. 5. Write a user-defined function to calculate the factorial of a given number. 6. Write Python programs to define the classes and objects and call them. 7. Write Python programs to show the concept of inheritance.

Scheme of Assessment

Grade XI

Table 1: Exam Specification

Topic No.	Topics	Marks Distribution			Total Marks
		MCQs	CRQs	ERQs	
1.	Computing Systems	8	Total 9 Marks (3 CRQs)		17
2.	Computational Thinking & Algorithm	8	Total 3 Marks (1 CRQ)		11
3.	Programming Fundamentals (Python)	13	Total 6 Marks (2 CRQs)	7 Marks Choose any ONE from TWO	26
4.	Data and Analysis	3	Total 3 Marks (1 CRQ)		06
5.	Applications of Computer Science	7		7 Marks Choose any ONE from TWO	21
6.	Impacts of Computing	7			
7.	Digital Literacy	4			04
8.	Entrepreneurship in the Digital Age	FA	FA	FA	-
	Total	**50**	**21**	**14**	**85**
	Practical*				**15**
	Total				**100**

Grade XII

Table 2: Exam Specification

Topic No.	Topics	Marks Distribution			Total Marks
		MCQs	CRQs	ERQs	
9.	Computing Systems	7	Total 3 Marks (1 CRQ)		10
10.	Computational Thinking & Algorithm	8	Total 3 Marks (1 CRQ)		11
11.	Programming Fundamentals (Python)	10	Total 9 Marks (3 CRQs)	7 Marks Choose any ONE from TWO	26
12.	Data and Analysis	7	Total 3 Marks (1 CRQ)		10
13.	Applications of Computer Science	5		7 Marks Choose any ONE from TWO	22
14.	Impacts of Computing	7	Total 3 Marks (1 CRQ)		
15.	Digital Literacy	3			03
16.	Entrepreneurship in the Digital Age	3			03
	Total	**50**	**21**	**14**	**85**
	Practical*				**15**
	Total				**100**

Examination Structure and Practical Requirements for Grades XI and XII

Theory:

- 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.
- Table 1 and 2 contain the marks distribution for each topic.
- 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 of 3 hours and will consist of two parts: paper I and paper II.
- Paper I theory will consist of 50 compulsory, multiple choice items. These questions will involve four response options. The answer sheet for paper I will be provided separately.
- Paper II theory will carry 35 marks and consists of Constructed Response Questions (CRQs) and Extended Response Questions (ERQs). Each extended response question will be presented in an either/ or form.
- The booklet for paper II will serve as an answer script.

Practical:

- In each grade, a practical examination (Paper III) will be conducted separate from the theory paper and will consist of 15 marks.
- Practical examination (Paper III) will be based on the list of practical activities given in the examination syllabus. Schools may design their practical manuals based on these activities for teaching and learning purposes.
- A 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 (Paper III).
- Each school needs to equip its computer lab with updated IT equipment, internet, etc., according to the requirements of the practical activities. Each school will be responsible for making sure that each student is provided the opportunity to do the practical activities.

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 **Syed Muhammad Waqas, Specialist** in **Computer Science** at AKU-EB, for taking the subject **lead** during the entire process of revising the HSSC Computer Science syllabus.

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

- **Sobia Zeeshan**
  Principal Syllabus Reviewer
  Bai Virbaiji Soparivala Parsi High School, Karachi
- **Syed Manzoor Mehmood**
  Principal Syllabus Reviewer
  Aga Khan Higher Secondary School, Karimabad, Karachi & Shahwilayat Public School (Campus II), Karachi
- **Syed Owais Ahmed**
  Principal Syllabus Reviewer
  Habib Public School, Karachi
- **Asharib Ali**
  Panellist
  Sir Adam Jee Institute of Management and Sciences, Karachi
- **Hussain Zohaib**
  Panellist
  Saifiyah Boys High School & Saifiyah Higher Secondary College, Karachi
- **Irfan Abdullah**
  Panellist
  Aga Khan Higher Secondary School, Karimabad, Karachi
- **Rana Farooqui**
  Panellist
  The Mama Parsi Girls' Secondary School
- **Sumaira Farhan**
  Panellist
  PECHS Girls' School, Karachi

We also thank the following **post-revision reviewers** for their feedback on the relevance of the content, skills, and resources of the syllabus:

- **Muhammad Umair Ayub**
  Non-Panellist
  IBAGRADS Intermediate Program, N. Nazimabad, Karachi
- **Syed Faizan Fahim**
  Non-Panellist
  The Taurus School, 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.

- **Mr. Saqib Noor**
  Lecturer, Computer Science Department, Aligarh Institute of Technology, Karachi

In-house Team

- **Mentor and Advisor:**
  Dr Naveed Yousuf
  CEO, AKU-EB
- **Operational Advisor:**
  Hanif Shariff
  Director, AKU-EB
- **Syllabi Review Lead:**
  Raabia Hirani
  Former Manager, Curriculum Development
- **Syllabi Review Facilitators:**
  Dur Nasab, Associate, Curriculum Development
  Mahrukh Jiwa, Specialist, Middle School Programme
- **Internal Reviewer:**
  Zain-ul-Muluk, Manager, Examination Development
  Afreen Kanwal, Lead Specialist, Examination Development
- **Learning Resources Reviewer:**
  Ali Bijani, Manager, Teacher Support, and team
- **Assessment Reviewer:**
  Munira Muhammad, Lead Specialist, Assessment
- **Administrative Support:**
  Raheel Sadruddin, Manager, Administration, and team
- **School Coordination Support:**
  Danish Hussain, Senior Manager, Operations**,** and team
- **Syllabi Feedback Data Analysts:**
  Muhammad Faheem, Lead Specialist, Assessment
  Muhammad Kamran Afzal, Specialist, Assessment
- **Communications and Design Support:**
  Shanoz Aqnazarbekova, Manager, Communications
  Hatim Yousuf, Specialist, Communications
- **Quality Assurance:**
  Malik Azam, Manager and team
- **Data Compilation and Formatting:**
  Shamsa Farzand Ali, Former Assistant, AKU-EB
  Ali Jumani, Assistant, AKU-EB
  Riyan Ali, Assistant, AKU-EB
  Akber Hashmani, Specialist, Assessment, AKU-EB

Aga Khan University Examination Board

Block-C, IED-PDC, 1-5/B-VII, Federal B Area,
Karimabad, Karachi, Pakistan - 75950

examinationboard.aku.edu
AKUEBOfficial
akuexamboard
AKUEBOfficial

AKUEBOfficial
linkedin.com/school/akueb
examination.board@aku.edu
+92 21 3682 7011-8