آغـــاخان یونیورسٹی ایگزامینیشن بورڈ

AGA KHAN UNIVERSITY EXAMINATION BOARD

Pacing Guide for Teachers

Computer Science

Grade X

Theory

Number of weeks: 28

Number of periods per week: 5

Key Textbook: Textbook of Computer Science Grade 10,

National Book Foundation as Federal Textbook Board Islamabad

Teacher Developer(s): Sobia Zeeshan and Najmus Sahar

Institution(s): Bai Virbaiji Soparivala Parsi High School and SMIU Model School, Karachi

Topic		Total Periods
7. Algorithms and Flowcharts		18

Sub-Topic	Range of SLOs	Periods (40 mins)
7.1 Understanding the Problem	7.1.1	6
7.2 Algorithm	7.2.1- 7.2.3	6
7.3 Flowchart	7.3.1- 7.3.5	6

Web Resource

https://www.bbc.co.uk/bitesize/guides/z3bq7ty/revision/3

Suggested Activities and/or Formative Assessment Guide

Activity 1

Ask students to create flowchart for a simple task: Students can start by creating a flowchart for a simple task, such as making a cup of tea or getting dressed in the morning. This will help them understand the basic elements of a flowchart, such as decision points and action steps.

Activity 2

Practice: Convert an algorithm to a flowchart.

Activity 3

Collaborative Learning: Collaborate on a group project: Students can work together on a group project, such as Using Visual Studio to practice flowcharts. This will help them practice their communication and collaboration skills, as well as their algorithm and flowchart designing skills. Their teamwork skills will be improved while providing a deeper understanding of the concepts.

Activity 4

Group Project: Teachers can assign a group project that requires students to work collaboratively to develop an algorithm and flowchart for a specific task. This will help students practice their communication and collaboration skills while also developing their algorithm and flowchart skills.

Activity 5

Individual Project: Teachers can assign an individual project that requires students to develop an algorithm and flowchart for a complex problem. This will allow students to practice their problem-solving skills and demonstrate their understanding of algorithm and flowchart principles.

Activity 6

Presentation: Students can present their algorithm and flowchart assignments to the class, explaining the reasoning behind their decisions and highlighting any challenges they encountered. This will help students develop their public speaking skills and provide an opportunity for peer feedback.

Activity 7

Error Correction: Teachers can provide students with a flowchart or algorithm that contains errors and ask them to identify and correct the errors. This will help students develop their critical thinking and problem-solving skills, as well as their ability to recognise common mistakes in algorithm and flowchart development.

Topic	**Total Periods**
8. Programming in C Language	16

Sub-Topic	Range of SLOs	Periods (40 mins)
8.1 Introduction to Programming Languages	8.1.1-8.1.3	3
	8.1.4-8.1.5	3
8.2 Programming Environment	8.2.1-8.2.2	3
8.3 Programming Basics	8.3.1-8.3.3	3
8.4 Constants and Variables	8.4.1-8.4.5	4

Learning Resource

- IT Series Revised & Updated Computer Science by Tasleem Mustafa Tariq Mahmood by New Kitab Markaz, Pakistan

Web Resource

https://www.w3schools.com/cpp/default.asp

Suggested Activities and/or Formative Assessment Guide

Activity 1

Install a C compiler such as Dev C++ or use any online compiler.

Activity 2

Declare and initialize Variables: Students can practice declaring and initialising variables using C compiler.

Activity 3

Write a "Hello World" Program: Students can write a simple "Hello World" program in C to learn the basics of the programming environment. This program will print the message "Hello, World!" to the screen and Practice IDE tools.

Activity 4

Peer Programming Time: Students can work in peers to write a C program that uses variables and constants. One student can write the code while the other provides feedback and suggestions. They can switch roles periodically to practice their programming and collaboration skills. Abilities, and communication skills. This activity will help students prepare for real-world job interviews and learn from feedback.

Activity 5

Code Review: Students can form groups and review each other's code for variables and constants usage. They can provide feedback on ways to improve the code and make it more efficient.

Activity 6

Debugging Challenge: Teachers can provide a C program with errors, and students can work together in groups to identify and fix the errors. This activity will help students practice their debugging skills and learn from their peers.

Activity 7

Variable Declaration Game: In this activity, students can form teams and take turns declaring variables of different data types. Each team must come up with a unique variable name and assign it a value. The team with the most unique and creative variable names wins the game.

Activity 8

Individual Coding Assignment: Teachers can assign an individual coding assignment that requires students to use variables and constants to solve a specific problem or task. Teachers can provide feedback on each student's submission and help them identify areas where they need improvement.

Activity 9

Code Analysis: Teachers can provide a C program that uses variables and constants and ask students to analyze the code and provide feedback on its structure, readability, and efficiency. This activity will help students develop their critical thinking and evaluation skills.

Activity 10

Multiple-Choice Quiz: Teachers can create a multiple-choice quiz that assesses students' understanding of the C programming environment, variable declaration, and constant usage. The quiz can cover topics such as data types, scope, and initialization. This activity will help students test their knowledge and identify areas where they need further study.

Topic	Total Periods
9. Fundamentals of Input and Output Handling in C Programming	18

Sub-Topic	Range of SLOs	Periods (40 mins)
9.1 Input and Output Functions in C Programming	9.1.1-9.1.5	3
9.2 Operators in C Language	9.2.1-9.2.2	3
	9.2.3-9.2.4	3
	9.2.5-9.2.6	3
	9.2.7-9.2.8	3
	9.2.9-9.2.11	3

Learning Resource

- IT Series Revised & Updated Computer Science by Tasleem Mustafa Tariq Mahmood by New Kitab Markaz, Pakistan

Web Resources

https://www.w3schools.com/cpp/default.asp

https://sabaq.pk/book-page.php?b=f&c=10&s=cs

Suggested Activities and/or Formative Assessment Guide

Activity 1

Basic input/output: Students can start by writing a simple C program that uses print and scan functions to take input from the user and display output on the console. They can practice using different format specifiers to display output in different formats.

Activity 2

Formatted Output: Students can write a C program that uses the print function to display formatted output using escape sequence characters. They can practice formatting output for different data types like integers, floating-point numbers, and strings using different types of operators.

Activity 3

Coding Challenge: Teachers can create a coding challenge that requires students to use input/output functions and operators to solve a specific problem or task. The challenge can have multiple levels of difficulty, and students can work individually or in groups to complete the challenge. This activity will help students test their knowledge and problem-solving skills.

Activity 4

Mock interview: Students can participate in a mock interview where they are asked to write a C program that uses input/output functions and operators to solve a specific problem. The interviewer can provide feedback on their coding skills, problem-solving abilities, and communication skills. This activity will help students prepare for real-world job interviews and learn from feedback.

Activity 5

Quiz: Create a quiz that tests students' knowledge of input/output functions and operators in C programming. This can include questions related to syntax, formatting, and usage. This will help students assess their understanding of the topic and identify areas for improvement.

Activity 6

Build a C program that takes attributes for a form to register students for a "summer vacation course."

Topic	Total Periods
10. Control Structure	18

Sub-Topic	Range of SLOs	Periods (40 mins)
10.1 Control Structure	10.1.1-10.1.2	3
	10.1.3-10.1.4	3
	10.1.5-10.1.6	3
	10.1.7-10.1.8	4
	10.1.9-10.1.11	5

Learning Resource

- IT Series Revised & Updated Computer Science by Tasleem Mustafa Tariq Mehmood by New Kitab Markaz, Pakistan

Web Resource

http://programmingclangauge.weebly.com/control-structure.html

Suggested Activities and/or Formative Assessment Guide

Activity 1

Assign students a programming task that requires them to use any control structure from if-else statements and switch statements once done, ask them to replace the control structure with the other type. This will help students practice applying control structures in real-world scenarios, selecting the most appropriate structure for any given situation.

Activity 2

Collaborative Coding: Pair students up and have them work together to write a program that uses control structures to solve a specific problem or task. This will help students practice collaboration and communication skills, as well as improve their programming skills.

Activity 3

Quiz: Create a quiz that tests students' knowledge of control structures in C programming. This can include questions related to syntax, formatting, and usage. This will help students assess their understanding of the topic and identify areas for improvement.

Activity 4

Code review: Provide students with a code snippet that contains control structures and ask them to review the code to identify any errors or areas for improvement. This will help students develop their ability to analyze and evaluate code.

Topic	Total Periods
11. Loop Structure	17

Sub-Topic	Range of SLOs	Periods (40 mins)
11.1 Loop Structure	11.1.1 -11.1.2	3
	11.1.3 -11.1.4	3
	11.1.5 -11.1.6	3
	11.1.7-11.1.8	4
	11.1.9	4

Learning Resource

- IT Series Revised & Updated Computer Science by Tasleem Mustafa Tariq Mehmood by New Kitab Markaz, Pakistan

Web Resource

https://www.w3schools.com/cpp/default.asp

Suggested Activities and/or Formative Assessment Guide

Activity 1

Writing programs: Assign students a programming task that requires them to use loop structures such as for loops, while loops, and do-while loops to solve a specific problem or task. This will help students practice applying loop structures in real-world scenarios and improve their problem-solving skills.

Activity 2

Code Tracing: Provide students with a C program that contains loop structures and ask them to trace the code to identify how the loop works and what the output will be. This will help students improve their understanding of loop structures and their ability to analyze code.

Activity 3

Pair Programming: Pair students up and have them work together to write a program that uses loop structures to solve a specific problem or task. This will help students practice collaboration and communication skills, as well as improve their programming skills. Code review: Ask students to exchange their code with a peer and provide feedback on how to improve the readability, efficiency, and reliability of the code. This will help students develop their critical thinking and evaluation skills.

Activity 4

Quiz: Create a quiz that tests students' knowledge of loop structures in C programming. This can include questions related to syntax, formatting, and usage. This will help students assess their understanding of the topic and identify areas for improvement.

Activity 5

Coding Challenge: Provide students with a coding challenge that requires them to use loop structures to solve a complex problem or task. This will help students apply their knowledge to real-world scenarios and practice their problem-solving skills.

Activity 6

Debugging Exercises: Provide students with a C program that contains errors related to loop structures and ask them to debug the code. This will help students practice problem-solving and error correction skills, as well as improve their understanding of loop structures.

Topic		Total Periods
12. Computer Logic and Gates		20

Sub-Topic	Range of SLOs	Periods (40 mins)
12.1 Data Representation	12.1.1	2
12.2 Logic Gates	12.2.1-12.2.2	3
	12.2.3-12.2.4	3
	12.2.5	3
	12.2.6	3
	12.2.7	3
12.3 Karnaugh Map (K-Map)	12.3.1	3

Learning Resource

- IT Series Revised & Updated Computer Science by Tasleem Mustafa Tariq Mehmood by New Kitab Markaz, Pakistan

Web Resource

Suggested Activities and/or Formative Assessment Guide

Activity 1

Truth Table Creation: Assign students a logic expression and ask them to create a truth table for it using logic gates. This will help students practice applying logic gates in real-world scenarios and improve their problem-solving skills.

Activity 2

Circuit Design: Provide students with a circuit diagram and ask them to identify which logic gates are being used in the design. This will help students improve their understanding of logic gates and their ability to analyse circuits.

Activity 3

Provide inputs for available gates in a logic diagram and ask for the resultant in a specified given situation.

Activity 4

Give a Logical expression/equation ask students to create a logic circuit.

Activity 5

Group Circuit Design

Divide students into groups and have them work together to design a circuit using logic gates to solve a specific problem or task. This will help students practice collaboration and communication skills, as well as improve their understanding of logic gates and circuit design.

Activity 6

Peer Feedback

Ask students to exchange their circuit designs with a peer and provide feedback on how to improve the readability, efficiency, and reliability of the circuit. This will help students develop their critical thinking and evaluation skills.

Activity 7

Give multiple logical expressions/equations and ask students to identify the correct expression for a given logic circuit.

Activity 8

Quiz: Create a quiz that tests students' knowledge of logic gates and circuit design. This can include questions related to logic gates, truth tables, and circuit diagrams. This will help students assess their understanding of the topic and identify areas for improvement.

Activity 9

Circuit Simulation: Provide students with a circuit simulation program and ask them to design and test a circuit using logic gates. This will help students apply their knowledge to real-world scenarios and practice their problem-solving skills.

Activity 10

Logic Gate Matching: Provide students with a list of logic expressions and a list of logic gates and ask them to match the expressions to the appropriate gates. This will help students practice identifying which gate is appropriate for a specific logic expression.

Topic	Total Periods
13. Office Automation II (MS Excel 2007 or Above)	12

Sub-Topic	Range of SLOs	Periods (40 mins)
13.1 Spreadsheet	13.1.1	2
	13.1.2	2
	13.1.3	2
	13.1.4	2
	13.1.5	4

Web Resource

https://support.microsoft.com/en-au/office/excel-video-training-9bc05390-e94c-46af-a5b3-d7c22f6990bb

Suggested Activities and/or Formative Assessment Guide

Activity 1

Data entry and Formatting: Assign students a dataset and ask them to enter and format the data in a spreadsheet program such as Microsoft Excel or Google Sheets. This will help students practice data entry and formatting skills.

Activity 2

Formula Creation: Provide students with a sample dataset and ask them to create formulas to perform calculations and analysis on the data. This will help students practice using formulas in spreadsheets and improve their problem-solving skills.

Activity 3

Ask students to highlight specific datasets according to the provided constraints using conditional formatting.

Activity 4

Spreadsheet Analysis: Divide students into groups and provide them with a complex dataset. Ask them to work together to analyze the data and create charts and graphs to visualize the results. This will help students practice collaboration and communication skills, as well as improve their data analysis and presentation skills.

Activity 5

Budgeting Project: Ask students to work in pairs or small groups to create a budget using a spreadsheet program. They should include income, expenses, and savings goals, and use formulas to calculate totals and balances. This will help students develop their financial literacy skills, as well as their spreadsheet skills.

Activity 6

Spreadsheet Project: Provide students with a real-world scenario that requires the use of spreadsheets, such as tracking expenses for a business or creating a project timeline. Ask students to use a spreadsheet program to complete the project and provide feedback on their work. This will help students apply their knowledge to real-world scenarios and practice their problem-solving skills.

Activity 7

Spreadsheet Design Challenge: Provide students with a list of requirements for a spreadsheet, such as specific formulas, formatting, and data entry requirements. Ask students to design the spreadsheet to meet the requirements and explain their design choices. This will help students develop their critical thinking and evaluation skills, as well as their spreadsheet skills.

Topic   **Total Periods**

14. Computer Security and Ethics   21

Sub-Topic	Range of SLOs	Periods (40 mins)
14.1 Introduction to Computer Security	14.1.1	2
	14.1.2	2
	14.1.3	2
	14.1.4	2
14.2 Computer Viruses	14.2.1	2
	14.2.2	2
	14.2.3	2
14.3 Authentication Mechanisms	14.3.1	2

	14.3.2	3
	14.3.3	2

Web Resource

Suggested Activities and/or Formative Assessment Guide

Activity 1

Research Paper: Ask the students to write a research paper on a specific computer virus or malware. This activity will help students understand the intricacies of viruses and how they can be a threat to computer security.

Activity 2

Case Study Analysis: Provide students with case studies of computer security breaches that occurred due to poor authentication mechanisms or virus attacks. Ask them to analyze the case studies and identify the root cause of the problem.

Activity 3

Presentation: Divide the class into groups and ask each group to create a presentation on a specific topic related to computer security, viruses, and authentication mechanisms. This activity will help students develop their presentation and research skills.

Activity 4

Express in your own words, what vulnerabilities a recent virus may bring that could not be a characteristics of an old virus program.

Activity 5

Class Discussion: Organise a class discussion on a specific topic related to computer security, viruses, and authentication mechanisms. This activity will help students understand different perspectives and viewpoints.

Activity 6

Reflection: Ask students to reflect on their learning throughout the course and write a reflection paper. This activity will help students evaluate their understanding of the topic and identify areas where they need to improve.

Note: This teacher-led pacing guide has been developed for AKU-EB affiliated schools to facilitate them by

- ensuring smooth transition of a school's academic year.
- ensuring curricular continuity in schools.
- predicting the time and pace of syllabi implementation.

This document also contains **suggested activities and/or formative assessments** that may enhance the learning experience. Please note that these activities are meant to serve as suggestions. As educators, you have the flexibility and autonomy to adapt and modify them to best suit the needs of your students and the dynamics of your classroom.

You are advised to use an ad-blocker while accessing the websites and web resources. In case any website is not functional for any reason, you may inform us at examination.board@aku.edu for an alternative or search material via any search engine.