



**Computer Science - Theory Progress Report 2019-20**

<b>Student Name: Arjun Raj</b>		<b>Teacher: Ashok B</b>				
<b>Learning Objectives</b>	<b>Term I (July - October)</b>			<b>Term II (November - March)</b>		
	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>
<b>Data representation</b>						
<b>Binary systems</b>						
Recognise the use of binary numbers in computer systems			✓			
Convert positive denary integers into binary and positive binary integers into denary			✓			
Show understanding of the concept of a byte and how the byte is used to measure memory size			✓			
Use binary in computer registers for a given application (such as in robotics, digital instruments and counting systems)			✓			



<b>Hexadecimal</b>						
Represent positive numbers in hexadecimal notation			✓			
Show understanding of the reasons for choosing hexadecimal notation to represent numbers			✓			
Convert positive hexadecimal integers to and from denary			✓			
Convert positive hexadecimal integers to and from binary			✓			
Represent numbers stored in registers and main memory as hexadecimal			✓			
Identify current uses of hexadecimal numbers in computing, such as defining colours in Hypertext Markup Language (HTML), Media Access Control (MAC) addresses, assembly languages and machine code, debugging			✓			
<b>Data storage</b>						
Show understanding that sound (music), pictures, video, text and numbers are stored in different formats			✓			
Identify and describe methods of error detection and correction, such as parity checks, check digits, checksums and Automatic Repeat reQuests (ARQ)			✓			
Show understanding of the concept of Musical Instrument Digital Interface (MIDI) files, JPEG files, MP3 and MP4 files			✓			
Show understanding of the principles of data compression (lossless and lossy) applied to music/ video, photos and text files			✓			



<b>Communication and Internet technologies</b>						
<b>Data transmission</b>						
Show understanding of what is meant by transmission of data			✓			
Distinguish between serial and parallel data transmission			✓			
Distinguish between simplex, duplex and half-duplex data transmission			✓			
Show understanding of the reasons for choosing serial or parallel data transmission			✓			
Show understanding of the need to check for errors			✓			
Explain how parity bits are used for error detection			✓			
Show understanding of the use of serial and parallel data transmission, in Universal Serial Bus (USB) and Integrated Circuit (IC)			✓			
<b>Security aspects</b>						
Show understanding of the security aspects of using the Internet and understand what methods are available to help minimise the risks			✓			
Show understanding of the Internet risks associated with malware, including viruses, spyware and hacking			✓			
Explain how anti-virus and other protection software helps to protect the user from security risks			✓			



<b>Internet principles of operation</b>						
Show understanding of the role of the browser			✓			
Show understanding of the role of an Internet Service Provider (ISP)			✓			
Show understanding of what is meant by hypertext transfer protocol (http and https) and HTML			✓			
Distinguish between HTML structure and presentation			✓			
Show understanding of the concepts of MAC address, Internet Protocol (IP) address, Uniform Resource Locator (URL) and cookies			✓			
<b>Hardware and software</b>						
<b>Logic gates</b>						
Use logic gates to create electronic circuits			✓			
Understand and define the functions of NOT, AND, OR, NAND, NOR and XOR (EOR) gates, including the binary output produced from all the possible binary inputs			✓			
Draw truth tables and recognise a logic gate from its truth table			✓			
Recognise and use the standard symbols used to represent logic gates (NOT, AND, OR, NAND, NOR, XOR)			✓			
Produce truth tables for given logic circuits			✓			



Produce a logic circuit to solve a given problem or to implement a given written logic statement			✓			
<b>Computer architecture and the fetch-execute cycle</b>						
Show understanding of the basic Von Neumann model for a computer system and the stored program concept			✓			
Describe the stages of the fetch-execute cycle, including the use of registers and buses			✓			
<b>Input devices</b>						
Describe the principles of operation (how each device works) of these input devices: 2D and 3D scanners, barcode readers, Quick Response (QR) code readers, digital cameras, keyboards, mice, touch screens, interactive whiteboards, microphones			✓			
Describe how these principles are applied to real-life scenarios, for example: scanning of passports at airports, barcode readers at supermarket checkouts, and touch screens on mobile devices			✓			
Describe how a range of sensors can be used to input data into a computer system, including light, temperature, magnetic field, gas, pressure, moisture, humidity, pH and motion			✓			
Describe how these sensors are used in real-life scenarios, for example: street lights, security devices, pollution control, games, and household and industrial applications			✓			
<b>Output devices</b>						
Describe the principles of operation of the following output devices: inkjet, laser and 3D printers; 2D and 3D cutters; speakers and headphones; actuators; flat-panel display screens, such as Liquid Crystal			✓			



Display (LCD) and Light-Emitting Diodes (LED) display; LCD projectors and Digital Light Projectors (DLP)						
Describe how these principles are applied to real-life scenarios, for example: printing single items on demand or in large volumes; use of small screens on mobile devices			✓			
<b>Memory, storage devices and media</b>						
Show understanding of the difference between: primary, secondary and off-line storage and provide examples of each			✓			
Describe the principles of operation of a range of types of storage device and media including magnetic, optical and solid state			✓			
Describe how these principles are applied to currently available storage solutions, such as SSDs, HDDs, USB flash memory, DVDs, CDs and Blu-ray discs			✓			
Calculate the storage requirement of a file			✓			
<b>Operating systems</b>						
Describe the purpose of an operating system			✓			
Show understanding of the need for interrupts			✓			
<b>High- and low-level languages and their translators</b>						
Show understanding of the need for both high-level and low-level languages			✓			
Show understanding of the need for compilers when translating programs written in a high-level language			✓			



Show understanding of the use of interpreters with high-level language programs			✓			
Show understanding of the need for assemblers when translating programs written in assembly language			✓			
<b>Security</b>						
Show understanding of the need to keep data safe from accidental damage, including corruption and human errors			✓			
Show understanding of the need to keep data safe from malicious actions, including unauthorised viewing, deleting, copying and corruption			✓			
Show understanding of how data are kept safe when stored and transmitted; use of passwords, both entered at a keyboard and biometric			✓			
Show understanding of how data are kept safe when stored and transmitted; use of firewalls, both software and hardware, including proxy servers			✓			
Show understanding of how data are kept safe when stored and transmitted; use of security protocols such as Secure Socket Layer (SSL) and Transport Layer Security (TLS)			✓			
Show understanding of how data are kept safe when stored and transmitted; use of symmetric encryption			✓			
Show understanding of the need to keep online systems safe from attacks including denial of service attacks, phishing, pharming			✓			
<b>Ethics</b>						
Show understanding of computer ethics, including copyright issues and plagiarism			✓			
Distinguish between free software, freeware and shareware			✓			



Show understanding of the ethical issues raised by the spread of electronic communication and computer systems, including hacking, cracking and production of malware			✓			
<b><u>Student's Status Report</u></b>						
	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>
Communication			✓			
Class Work		✓				
Home Work			✓			
Readiness for class		✓				
Understanding the concept and instructions			✓			
Class Participation		✓				
Independence			✓			
Presentation			✓			





---

## **Term I**

### **Content Covered**

All of the content for the 10th board has been completed by the end of 9th grade. We are spending time in 10th grade revising the topics and writing tests.

### **Comments:**

#### **Areas of Strength**

Arjun is naturally very strong in this subject. He has understood pretty much all of the concepts well and is in a good place to score a high grade in this subject. It has been a pleasure to teach him.

#### **Suggestion for Growth**

Since we have completed the syllabus in 9th grade itself and are spending time during teen circles revising the topics where Arjun is not able to be present, it is important for him to revise by himself at home. It will also be good if he is able to attend the teen circle classes from time to time as well so he can bring up any doubts he has for clarification.



**Computer Science - Practical Progress Report 2019-20**

<b>Student Name : Arjun Raj</b>	<b>Teacher: Mamatha Pramod</b>					
<b>Learning Objectives</b>	<b>Term I (July - October)</b>			<b>Term II (November - March)</b>		
	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>
<b>Algorithm design and problem-solving</b>						
<b>Problem-solving and design</b>						
Show understanding that every computer system is made up of sub-systems, which in turn are made up of further sub-systems		✓				
Use top-down design, structure diagrams, flowcharts, pseudocode, library routines and subroutines		✓				
Work out the purpose of a given algorithm			✓			
Explain standard methods of solution			✓			
Suggest and apply suitable test data			✓			
Understand the need for validation and verification checks to be made on input data			✓			



Use trace tables to find the value of variables at each step in an algorithm			✓			
Identify errors in given algorithms and suggest ways of removing these errors			✓			
Produce an algorithm for a given problem (either in the form of pseudocode or flowchart)			✓			
Comment on the effectiveness of a given solution			✓			
<b>Pseudocode and flowcharts</b>						
Understand and use pseudocode for assignment, using ←			✓			
Understand and use pseudocode, using the following conditional statements: IF ... THEN ... ELSE ... ENDIF			✓			
Understand and use pseudocode, using the following conditional statements: CASE ... OF ... OTHERWISE ... ENDCASE			✓			
Understand and use pseudocode, using the following loop structures: FOR ... TO ... NEXT			✓			
Understand and use pseudocode, using the following loop structures: REPEAT ... UNTIL			✓			
Understand and use pseudocode, using the following loop structures: WHILE ... DO ... ENDWHILE			✓			
Understand and use pseudocode, using the following commands and statements: INPUT and OUTPUT (e.g. READ and PRINT)			✓			
Understand and use pseudocode, using the following commands and statements: totalling (e.g. Sum ← Sum + Number)			✓			



Understand and use pseudocode, using the following commands and statements: counting (e.g. Count ← Count + 1)			✓			
Understand and use standard flowchart symbols to represent the above statements, commands and structures			✓			
<b>Programming</b>						
<b>Programming concepts</b>						
Declare and use variables and constants			✓			
Understand and use basic data types: Integer, Real, Char, String and Boolean			✓			
Understand and use the concepts of sequence, selection, repetition, totalling and counting			✓			
Use predefined procedures/functions			✓			
<b>Data structures; arrays</b>						
Declare and use one-dimensional arrays, for example: A[1:n]			✓			
Show understanding of the use of one-dimensional arrays, including the use of a variable as an index in an array			✓			
Read or write values in an array using a FOR ... TO ... NEXT loop			✓			



---

<b>Databases</b>				
Define a single-table database from given data storage requirements			✓	
Choose and specify suitable data types			✓	
Choose a suitable primary key for a database table			✓	
Perform a query-by-example from given search criteria.			✓	



<b>Student's Status Report</b>						
	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>	<b>Beginning</b>	<b>In Progress</b>	<b>Mastery</b>
Communication			✓			
Class Work			✓			
Home Work			✓			
Readiness for class			✓			
Understanding the concept and instructions			✓			
Class Participation			✓			
Independence			✓			
Presentation			✓			



---

## **Term I**

### **Content Covered**

#### *Algorithm: Pseudocode*

1. INPUT and OUTPUT statements
2. Conditional statements
  - 2.1. IF ... THEN ... ELSE ... ENDIF
  - 2.2. CASE ... OF ... OTHERWISE ... ENDCASE
3. Loop structures
  - 3.1. REPEAT ... UNTIL
  - 3.2. WHILE ... DO ... ENDWHILE
4. Standard actions
  - 4.1. Totalling (e.g.  $\text{Sum} \leftarrow \text{Sum} + \text{Number}$ )
  - 4.2. Counting (e.g.  $\text{Count} \leftarrow \text{Count} + 1$ )
5. Using trace tables
6. Identifying and correcting errors
7. The effectiveness of a solution

#### *Python programming*

1. Declaration and use of:
  - 1.1. Variables
  - 1.2. Constants
  - 1.3. Basic data types: Integer/Real/Char/String/Boolean
2. How to use:
  - 2.1. Sequence
  - 2.2. Selection
  - 2.3. Repetition
  - 2.4. Totalling
  - 2.5. Counting
3. Use of predefined procedures/functions

### **Comments:**

#### **Areas of Strength**

Arjun effectively interprets and synthesizes information and always participates in class with eagerness and enthusiasm. He is an independent learner who constructively uses his class time. He has the ability to analyse, evaluate, make reasonable judgements and present conclusions. He is able to interpret algorithms and provide a solution to a problem.

#### **Suggestion for Growth**

Planning and compiling his answers before attempting to answer will avoid scribbling or scratching thereby making the answer sheet look presentable in the board exams.

