## Section1: Theory of Computer Science

### August
- **1.1 Data Representation**
  - 1.1.1 Binary Data
  - 1.1.2 Hexadecimal
  - 1.1.3 Data Storage

### September
- **1.3 Hardware**
  - 1.3.1 Logic gates
  - 1.3.3 Input device

### October
- **1.3 Hardware**
  - 1.3.4 Output devices

### November
- Revision

### December
- **Mid-Year Examination**

## Section2: Practical Problem-solving and Programming

### January
- **2.1 Algorithm Design and Problem Solving**
  - 2.1.1 Problem-Solving and Design

### February
- **2.2 Programming (continued)**
  - 2.1.2 Pseudocode

### March
- 2.2 Programming (continued)
  - 2.1.2 Pseudocode

### April
- Revision

### May
- Final Examination
### August

#### Section 1: Theory of Computer Science

1.1 Data Representation
   
   1.1.1 Binary Data
   1.1.2 Hexadecimal
   1.1.3 Data Storage

<table>
<thead>
<tr>
<th>Contents</th>
<th>Activities or Learning resources</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BINARY AND HEXADECIMAL NOTATION:</strong></td>
<td>The concepts will be demonstrated by solving questions on board.</td>
</tr>
</tbody>
</table>
| Conversion of Binary numbers to and from Denary  
  • Perform conversions of binary and hexadecimal number to and from denary system |                                                                                                   |
| Conversion of Hexadecimal numbers to and from Denary  
  • Perform conversions of binary and hexadecimal number to and from denary system |                                                                                                   |
| Use of hexadecimal numbers:       | Binary counter – for example: [www.mathsisfun.com/binary-decimal-hexadecimal-converter.html](http://www.mathsisfun.com/binary-decimal-hexadecimal-converter.html) |
  • Identify the use of Hexadecimal in representing colors in HTML, representing MAC address, memory dumps, web address, assembly and machine code |                                                                                                   |
| Use of binary number system       | Binary numbers at Computer Studies Unplugged: [http://csunplugged.org/binary-numbers](http://csunplugged.org/binary-numbers) |
  • For IP addressing of resources on the internet  
  • Discuss and understand the concept of byte; also learn to calculate the memory requirements in form of Kb, Mb, Gb, Tb. |                                                                                                   |
  • Identify common file standards: JPG, GIF, PDF, MP3, MPEG, and MIDI. |                                                                                                   |
  Types media files: MIDI & MP3, MP4, JPEG | Simple comparisons at [www.bbc.co.uk/schools/gcsebitesize/ict/hardware/1datastoragerev2.shtml](http://www.bbc.co.uk/schools/gcsebitesize/ict/hardware/1datastoragerev2.shtml) |
| Data compression: Lossless and lossy compression applied to music/video, photos, and text files  
  • Understand the ways to detect and then correct errors: Parity check, check digits, checksums, ARQ (Automatic Repeat Request)  
  • Differentiate between lossless & lossy compression | Useful reinforcement material: [http://computer.howstuffworks.com/bytes.htm](http://computer.howstuffworks.com/bytes.htm) |
| For Practice: Past paper questions on Binary data:  
  Q15: 7010_s12_qp_11  
  Q11: 7010_w11_qp_11 |                                                                                                   |
| Practice questions to convert to and from GB, MB, KB |                                                                                                   |
| Book Reference:                   |                                                                                                   |
  Unit 2 |                                                                                                   |
| Handout: Will be shared with students |                                                                                                   |
### Contents

**LOGIC GATES**

- 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 recognize a logic gate from its truth table.
- Produce truth tables for given logic circuits.
- Produce a logic circuit to solve a given problem.
- Implement a logic circuit using 'NAND' gates only, by replacing each of the basic logic gate into its equivalent NAND gate circuit.

**INPUT DEVICES**

- Describe the principles of operation of a range of input devices including:
  - Scanners, barcode readers, digital cameras (including CCTV cameras), keyboards, mice, touch screens, microphones, QR code, RFID tags, sensors, Interactive whiteboards etc.
- Describe how these principles are applied to real-life scenarios, for example:
  - Use of Barcodes, QR codes, RFID tags in supermarket
  - Use of 2D, 3D scanners at airports.
  - Use of microphone in Voice Recognition systems
- 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/acidity/alkalinity and motion/infra-red.
- Describe how these sensors are used in real-life scenarios; for example:
  - Burglar alarm system
  - Anti lock braking system
  - Weather forecasting
  - Street lights controlling system

### Activities or Learning resources

**Book Reference:** Unit#11, Pg 274 (from Chris Leadbetter)  
Unit#3, (from David Watson and Helen Williams)

**Resource:** A handout based on the questions from past exams will be shared with the students.

Simple logic simulator using standard symbols:  

Downloadable logic gate simulator:  
[www.softpedia.com/get/Others/Home-Education/Logic-Gate-Simulator.shtml](http://www.softpedia.com/get/Others/Home-Education/Logic-Gate-Simulator.shtml)

and  
[www.logiccircuit.org/](http://www.logiccircuit.org/)

Websites such as  

**Book Reference:** ‘Unit# 3-Hardware’ Pg49-70

Illustrated notes on sensors:  

Notes on sensors:  
[www.bbc.co.uk/schools/gcsebitesize/ict/measurecontrol/Ocomputercontrolrev2.shtml](http://www.bbc.co.uk/schools/gcsebitesize/ict/measurecontrol/Ocomputercontrolrev2.shtml)

**Activity 5.6:** Pg# 104
October

Section1: Theory of Computer Science
1.3 Hardware
1.3.4 Output devices

Contents

OUTPUT DEVICES
- Describe the principles of operation of a range of output devices, including:
  - Inkjet, laser and 3D printers
  - 2D and 3D cutters
  - Speakers and headphones
  - Actuators (Buzzers, motors, valves)
  - Flat-panel display screens (LED, LCD)

- Describe how these principles are applied to real-life scenarios, for e.g.
  - Use of Inkjet printers for best off photos
  - Use of Laser printer for large volume printing
  - Use of 3D Printers, 2D & 3D Cutters in Automated industries.
  - Role of actuators in Control Applications.

Activities or Learning resources

Book Reference: ‘Unit# 3-Hardware’ Pg71-78

Websites such as:
- http://computer.howstuffworks.com/
- www.bbc.co.uk/schools/gcsebitesize/ict/hardware/1datastoragerev2.shtml

Websites such as:
- http://en.wikipedia.org/ which has entries for various related topics

Videos:
Short videos will be shown to further strengthen their concepts

November
Revision

Revision Tests
Practice worksheets
Assignments
Past Papers

December

Mid-Year Examination

January

Section2: Practical Problem-solving and Programming

2.1 Algorithm Design and Problem Solving
  2.1.1 Problem-Solving and Design
  2.1.2 Pseudocode
### Contents

**PROBLEM SOLVING:**

- Understand that every computer system is made up of subsystems.
  - Top-down design approach and discuss the advantages and disadvantage of top-down design approach
- Definition, purpose and testing of Algorithms
- Use of standard methods of solution.
- Application of suitable test data and know the basic data types: String, Integer, and Character and Boolean.
- Explain and apply test data: Normal data. Abnormal and extreme data.
- Identify errors in given algorithms and suggest ways of removing these errors.
  - Differentiate between logical and syntax error
- Dry running of Pseudocodes/flowcharts using trace tables
- Understand the need for validation and verification checks:
  - Range check, Length check/Limit check,
  - type check (character, numeric, alphanumeric) Consistency,
  - Format, Presence
  - check digits and checksum
- Produce an algorithm for a given problem
- Comment on the effectiveness of solution

### Activities or Learning resources

- An introduction to algorithmic thinking:
- PowerPoint presentation on flowcharts and program design:
  [http://staffweb.itsligo.ie/staff/bmulligan/Lectures/CSFrench/French08.ppt](http://staffweb.itsligo.ie/staff/bmulligan/Lectures/CSFrench/French08.ppt)
- Some self-checking flowchart exercises, with outline structure and available operations:
  [www.cimt.plymouth.ac.uk/projects/mepres/book8/bk8i1/bk8_1i2.htm](http://www.cimt.plymouth.ac.uk/projects/mepres/book8/bk8i1/bk8_1i2.htm)
- Pseudocode in ‘Absolute beginner’s guide to programming’:
  [http://books.google.co.uk/](http://books.google.co.uk/)
- Notes, quizzes and activities for data validation:
  [www.bbc.co.uk/schools/gcsebitesize/ict/databases/3datavalidationrev1.shtml](http://www.bbc.co.uk/schools/gcsebitesize/ict/databases/3datavalidationrev1.shtml)
  [www.klbict.co.uk/gcse/theory/5_3/5_3_3_valid_verif.htm](http://www.klbict.co.uk/gcse/theory/5_3/5_3_3_valid_verif.htm)
- Practice Question from past papers will be given for practice.
- Resource: Paper1 of 7010 from 2005 to 2014

### February & March

**Section2: Practical Problem-solving and Programming**

2.2 Programming (continued)

- 2.2.1 Programming concepts
### Contents

- Verification of data, which includes:
  - double entry
  - screen/visual check
  - parity check
  - Checksum.
- Basic data types: Integer, Real, Char, String and Boolean
- Declaration and use of variables and constants
- Use the concept of sequence, selection, repetition, totaling and counting to solve problems using:
  - Selection statements (IF...THEN...ELSE... & Case...OF...OTHERWISE...)
  - Repetition loops (FOR...NEXT, REPEAT...UNTIL, WHILE...DO)
- Use of Operators(+, -, *, /, ^, \&, <, >, <=, >=)

### Activities or Learning resources

- Some simple tasks in Scratch:  
  [www.teach-ict.com/programming/scratch/scratch_home.htm](http://www.teach-ict.com/programming/scratch/scratch_home.htm)
- Control programming:  
- Flowol website:  
- Some LOGO websites and ideas:  
  [www.mathcats.com/gallery/logodownloadinfo.html](http://www.mathcats.com/gallery/logodownloadinfo.html)

#### April Revision

<table>
<thead>
<tr>
<th>Contents</th>
<th>Activities or Learning resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision Tests</td>
<td>Some simple tasks in Scratch:</td>
</tr>
<tr>
<td>Practice worksheets</td>
<td>Control programming:</td>
</tr>
<tr>
<td>Assignments</td>
<td><a href="http://www.teach-ict.com/programming/scratch/scratch_home.htm">www.teach-ict.com/programming/scratch/scratch_home.htm</a></td>
</tr>
</tbody>
</table>

#### End of chapter questions:

- Activity 9.9 to 9.13 Pg#185
- Activity 10.4: Pg# 208
- Activity 10.5 : Pg#210
- Activity 10.6: Pg# 212
- End of chapter questions: Pg#216-218
- End of chapter question: Pg# 241

### May Final Exams