REPUBLIC OF RWANDA

MINISTRY OF EDUCATION
NATIONAL CURRICULUM DEVELOPMENT CENTRE (NCDC)
P.O. BOX 608 KIGALI
www.ncdc.gov.rw

COMPUTER SCIENCE CURRICULUM FOR COMPUTER SCIENCE ECONOMICS AND MATHEMATICS OPTION & MATHEMATICS PHYSICS AND COMPUTER SCIENCE OPTION

Kigali, February 2010

LIST OF PARTICIPANTS IN COMPUTER SCIENCE CURRICULUM DEVELOPMENT

1. **GAHIMA Charles:** Director General, NCDC (Coordinator)
2. **RUTAKAMIZE Joseph**: Director of Science and Arts Unit, NCDC (Supervisor)
3. **RWAMBUKA Aimable Protais** Acting Director of ICT Unit, NCDC (Supervisor)
4. **NYANGOMA Vincentia** : In Charge of ICT implementation in schools, NCDC
5. **NYIRANDAGIJIMANA Anathalie**: Curriculum Developer in charge of Pedagogical Norms, NCDC
6. **BAYONGWA Pacifique** : Teacher at ETP NYARUREMA
7. **NTAMUSHOBORA Peter Claver** : Teacher at SONRISE
8. **KURADUSENGE Benjamin** : Teacher at College ASPESKA
9. **UMURERWA Joyeuse**: Teacher at E.S. MUSAMVU

TABLE OF CONTENT

I. INTRODUCTION

The government's Vision 2020 and EDPRS set out ambitious plans to create a growing knowledge based economy hinged on a skilled workforce that can compete in the region and the wider international arena. In this connection, the curriculum and teaching practices should ensure that secondary school graduates are prepared for both labour market and higher education studies.

With the ICT curriculum for Ordinary level, learners have basic knowledge and skills that enable them to be familiar with the computer as an every day life tool. The computer science curriculum for Advanced level Science combinations is build on this prerequisite and it is intended to produce graduates who are able to use ICT skills for their social and economic well being.

This curriculum focuses on four courses: Introduction to computer, Computer programming, Maintenance and Database. Some courses are distributed across the whole Advanced level from Senior 4 to Senior 6 in order to establish a logical progression through the curriculum and to facilitate their mastery by learners.

II. METHODOLOGICAL NOTES

The teacher/facilitator should use different methods and techniques in computer science teaching. The computer science subject aims essentially to develop learners' practical skills. To this end, the teacher should focus on practical exercises rather than theoretical concepts that are quickly forgotten. Across the whole curriculum, several teaching and learning activities are suggested to the teacher, but he/she is free to think about others methods and techniques taking into consideration the teaching aids availability, the students' level and abilities. The teacher should use learner centred methods in order to develop learners' skills such as: Practical, Communication, Research, Problem solving, Observation, Creative, Innovative, Social skills, etc. Among these methods and techniques we have: work groups supplemented by individual homework, discussions, practical exercises, individual or cooperative research, observation of the concrete materials, etc.

III. EVALUATION APPROACH

Evaluation should be done to asses the attainment of the already set/ defined curriculum objectives. The teacher is urged to carry out the following forms of evaluation:

- Formative: this involves lesson, chapter evaluation to find out if the lesson or chapter objectives are attained;
- Summative: end of a term, year and level evaluation.

All these forms of evaluation should not only focus on theories but should assess students’ practical skills. Teachers should set practical tests that relate to scientific and daily life situations in order to test students’ reasoning and technical skills.

For National Exam:

- **Theoretical exam** should include: C programming language, C++ programming language, Visual Basic language, Algorithm, networking, operating system, database and maintenance
- **Practical exam** should include: C programming language, C++ programming language, Visual Basic language and Database

IV. REQUIRED EQUIPMENT

The required equipments to teach this course are the following:

- The availability of one or more computer laboratories. We recommend the ratio of 2 pupils per computer at most;
- The availability of relevant software specified as didactic materials;
- Internet connection for documentation and research.

V. GENERAL OBJECTIVES BY THE END OF ‘A’ LEVEL

By the end of senior 6, the learner will be able to:

- Design, install, maintain and administer a database (for small business)
- Maintain and assemble computers
- Design, maintain and administer a network (for small business)
- Install application programs
- Show the following skills in his every day life: Communication, Research, Practical Problem solving, Observation, Creative and Innovative skills.

VI. LIST OF COURSES AND DETAILED WEEKLY TIME ALLOCATION

Courses	**Number of periods per week**		
	S4	**S5**	**S6**
General introduction to computer	4	2	-
Maintenance	-	-	2
Computer Programming	3	2	3
Database	-	3	2
Total	**7**	**7**	**7**

SENIOR FOUR

1. List of courses and chapters

Course: Introduction to computer I (120 periods)
Chapter 1: General introduction to computer (24 periods)
Chapter 2: History of computer (8 periods)
Chapter 3: Computer System (22 periods)
Chapter 4: Computer Architecture (26 periods)
Chapter 5**:** Computer Arithmetic (40 periods)

Course: Computer Programming (90 periods)
Chapter 1: Algorithm (30 periods)
Chapter 2: C Programming Language (60 periods)

2. General objectives

By the end of senior 4, students should be able to:

- Analyse and suggest resolution to a given problem by illustrating succession of simple and accurate steps implicated in solving a given problem
- Draw a flowchart that shows steps followed in solving a problem
- Write a program in C programming language
- Describe computer system

3. Course 1: Introduction to computer I

Chapter 1: General Introduction to computer (24 periods)

- **Pre-requisites:** Basic computing
- **Teaching aids**: Personal computer with Windows XP or new version of Windows, Application software.
- **Skills to be developed**: Observation skills
- **General objective**: Identify basic functions of a Computer

Specific objectives	Content	Suggested teaching and learning activities
➢ Describe functions	Define a computer Computer function ✓ Input ✓ Processing ✓ Output	Teacher draws computer functional diagram to explain computer function
➢ Identify the main components of a computer	Computer components ✓ Hardware ✓ Software Computer Characteristics	The teacher shows computer's components in the laboratory

Chapter 2: History of computer (8 periods)

- **Pre-requisites**: General introduction to computers
- **Teaching aids**: Computer with Windows XP or new version of Windows, Transistor, Integrated Circuit, Internet(those who have not transistor, Integrated Circuit)
- **Skills to be developed**: Observation, research, social and communication skills
- **General objective**: Describe different generations of computers

Specific objectives	Content	Suggested teaching and learning activities
➢ Narrate the history of computer	History of a computer	**Work groups**: Internet research on history of a computer and presentation of results
➢ Distinguish different computer generation	Generations of a computer o 1st generation o 2nd generation o 3rd generation o 4th generation	

Chapter 3: Computer System (22)

- **Pre-requisites**: Function of a computer
- **Teaching aids**: computer with installed softwares
- **Skills to be developed**: research, social and communication skills
- **General objective**: Describe computer system and the role of computer in society

Specific objectives	Content	Suggested teaching and learning activities
The learner should be able to : ➢ Identify the components of computer system	**Computer system** ➢ Computer people system ➢ Hardware system ➢ Software system ➢ Dataware system	Demonstrate, explain computer system components
➢ Give the impact of computer in society	**Role of a computer** ➢ Medicine ➢ Education	Group work: discussion on the impact of computer in Rwandan society

	➢ Economic ➢ Communication ➢ Security ➢ Social	

Chapter 4: Computer Architecture (26 periods)

- **Pre-requisites**: Function of a computer
- **Teaching aids**: computer with its components
- **Skills to be developed**: Observation skills
- **General objective**:
  - Explain different families of computer
  - Differentiate type of memories and explain Hard disk Components

Specific objectives	Content	Suggested teaching and learning activities
List the different families of computers	**Classification of computers** ➢ Personal computer ➢ Mini computer ➢ Mainframe computer ➢ Super computer	Teacher shows computer of each family using concrete materials and drawings
Give the function of Central Processing Unit	**CPU(Central Processing Unit)** ➢ ALU(Arithmetic Logic Unit) ➢ CU(Control Unit) ➢ Registers	The teacher explains the function of each CPU components using drawings
Differentiate type of memories	**Memories** ➢ Definition	The teacher explains advantages and disadvantages of each type of memory

Identify the type and the characteristics of a hard disk	➢ Role of memory ➢ Features: o Capacity o Speed o Volatility ➢ Type of memories: o ROM memory o RAM memory The hard disk components ➢ Cylinder ➢ Platters ➢ Track ➢ Sector ➢ Configurations advised to put several peripherals IDE.	Exercise on how to calculate the hard disk capacity.

Chapter 5: Computer Arithmetic (40 periods)

- **Pre-requisites:** Algebra (Numeration system)
- **Teaching aids**: White/Blackboard, marker pen /chalk
- **Skills to be developed**: Reasoning skills
- **General objective**: To convert from one base to any base and use Boolean algebra and logic gates

Specific objectives	Content	Suggested teaching and learning activities
Convert from one base to another	**Introduction to coding** ➢ Decimal Base ➢ Binary Base ➢ Octal base ➢ Hexadecimal base o Binary operations o Decimal to binary conversion o Binary to decimal conversion o Hexadecimal to binary conversion o Binary to hexadecimal conversion o Converting to any base	Exercises on base conversion Exercises on how to simplify expression
Use Boolean algebra	➢ Boolean Algebra o Morgan's law o Associative law o Commutative law	
Use logical gates	➢ Logic gates o AND and NAND o OR and NOR o X-OR and N X-OR	

4. Course 2: Computer Programming

Chapter 1: Algorithm (30 periods)

- **Pre-requisites:** Basic concepts in mathematics and entrepreneurship for O'level
- **Teaching aids**: White/Blackboard, marker pen /chalk
- **Skills to be developed**: Analysis and reasoning skills
- **General objective**: To demonstrate steps to be followed in order to solve any mathematics, physics and economics problem
- **Methodological notes**:
  - This course of computer programming should focus on the other subjects within the combination (Mathematics, Physics and Economics).
  - Exercises on writing a program should be based on everyday life examples

Specific objectives	Content	Suggested teaching and learning activities
Define an algorithm	**Concept of algorithm** ➢ Definition of an algorithm ➢ Importance of an algorithm ➢ Characteristics ➢ Convention and terminology	The teacher gives an algorithm of the everyday life to illustrate all the steps of an activity
Explain the advantage of a variable in an algorithm	**Variables** ➢ Types of variables ➢ Assignment of the variables ➢ Temporary Variable	The teacher differentiates the container from the content; variable from the constant using examples The teacher makes the analogy between a variable and the container using examples
Explain how operators are used in an algorithm.	**Expressions and Operators** ➢ Numerical operators ➢ Alphanumeric operators ➢ Comparisons operators ➢ Logical operators (AND, OR, NOT)	The teacher uses tables of truth to demonstrate logical operators and gives exercises for application. The teacher uses operators to write a pseudo code and flow chart and gives exercises for application.

Distinguish the functions from Reading (input) and those of displaying (output).	**Reading and Writing** ➢ Reading function (input) ➢ Displaying function(output)	The teacher writes an algorithm using i/o functions in pseudo code and flow chart and gives exercises for application.
Give the situations of the use of various tests	**Tests** ➢ Structure of a test ➢ Conditions (if, if. .else, switch) ➢ Nested Ifs	The teacher writes an algorithm using the tests in pseudo code and flow chart and gives exercises for application.
Give the situation of the use of GOTO	➢ Go to(goto)	
Give the situations where various loops are used	**LOOPS** ➢ Loops (for, while, do while) ➢ Iterative Loops ➢ Loops in Loops	Give examples of loop and iterative loop. Write an algorithm using loops in pseudo code and flow chart.
Handle a table	**Use a table in Algorithm** ➢ Use of the Tables in Algorithm ➢ Dynamic Tables	Exercise on how to draw flowchart with Tables Exercise on how to write algorithms with Tables in pseudo code and flow chart

Chapter 2: C Programming Language (60 periods)

- **Pre-requisites:** Algorithm
- **Teaching aids:** Blackboard / Whiteboard, computer with ‘C’ compiler, chalk/marker
- **Skills to be developed**: Research, creative and innovation skills
- **General objective**: Write programs from an algorithm

Specific objectives	Content	Suggested teaching-learning activities
Define a program	**Concept of program** ➢ Introduction ➢ Define a program ➢ Evolution ➢ Write your first program in C ➢ Structure of C programs	The students find the evolution of C programming language using internet or different books.
Declare Variables and their type	**Variables and their type** ➢ Definition and initialization of a variable ➢ Assignment of value to variables char, int, long, short, float, double, double long, unsigned and signed. ➢ Variable scope ➢ The different memory space occupied by each variable type	The teacher gives a basic exercise to enable the students to test the effect on the result according to the type of variable. The teacher gives an exercise allowing the student to describe the overflow and underflow
Use the I/O functions	Scanf() and printf() functions	The teacher writes a small program that displays for instance ‘Hello class’ and gives exercises for application.
Declare Constants	**The difference between a constant and a variable** ➢ Use of a constant ➢ Declaration of a constant with const	The teacher writes a small program that accepts a value and displays it. Teacher gives exercises for application.
Use arithmetic operators	➢ Arithmetic operators “+”,” - “, “*”,”/“, “%”, “+=”,” - = ", “/=”, “*=”, “%=” and their priorities.	The teacher declares constant using the const and #define key words and gives exercises for

<table>
<tr><th></th><th></th><th></th></tr>
<tr><td></td><td></td><td>application.
The teacher writes programs where two values get involved in all arithmetic operators and gives exercises for application.
The teacher writes programs to show priority of one operator over another and gives exercises for application.</td></tr>
<tr><td>Use conditional expressions with comparison operators</td><td><b>Comparison operators</b>
“<”, “>”, “<=”, “>=”, “==”,”! = " and their use with If, if...else, switch</td><td>The teacher draws flowcharts to illustrate each program
The teacher writes programs to compare two or more values and gives exercises for application.</td></tr>
<tr><td>Use Logical operators</td><td><b>Logical operators</b>
“&&”, “||”,”! “and their use conditional expression
Priority of AND versus OR</td><td>The teacher writes a program using logical operators and gives exercises for application.
The teacher draws flowcharts to illustrate steps of each program</td></tr>
<tr><td>Use loops in c program</td><td><b>Loops</b>
➢ While
➢ Do...while
➢ For</td><td>The teacher writes programs using loops and gives exercises for application.</td></tr>
<tr><td>Use of continue and break</td><td>➢ Continue
➢ Break
➢ exit</td><td>The teacher writes programs including continue, break and exit keywords.
The teacher gives exercises for application.</td></tr>
<tr><td>Create and Use functions</td><td><b>Function</b>
➢ Name and syntax of a function
➢ Function without parameter.</td><td>The teacher writes a program including functions</td></tr>
</table>

	➢ functions call ➢ Creation of function ➢ Mathematical built-in functions (sqrt(),abs(),sinus, cosine, log()...etc)	The teacher writes program using mathematic built-in functions and gives exercises for application
Create and use one dimensional array	**One-dimensional array** ➢ Definition ➢ Declaration of one-dimensional array ➢ Use array	The teacher draws flowcharts, writes pseudo codes leading to write a given programs with arrays The teacher writes a program with arrays and gives exercises for application.

5. Lesson Distribution per term (12 weeks per term)

Course: Introduction to computer 1st term

Week	Specific objectives	Content	Periods
1-3	Describe computer functions	General Introduction to computer Computer function	**12**
4-7	Identify the main component of a computer	Computer components Computer Characteristics	**12**
7-8	Narrate the history of computer and Explain different computer generation	History of a computer	**8**
9-10	Identify the components of computer system	Computer system	**8**
11	Revision		**4**
12	Exams		**4**
Total			**48**

2nd Term

Week	Specific objectives	Content	Periods
1-3	Give the impact of computer to society	Role of computer in society	**12**
4-5	List the different families of computers	Classification of computers	**8**
7-10	Describe memories and their different use	Memories	16
11	Revision		**4**
12	Exams		**4**
Total			**48**

3rd term

Week	Specific objectives	Content	Periods
1-4	Convert from one base to another	Introduction to coding Bases	16
5-7	Use Boolean algebra	Boolean Algebra	**12**
8-10	Use logical gates	Logical gates	**12**
11	Revision		**4**
12	Exams		**4**
Total			**48**

Course: Computer Programming

1st term

Week	Specific objectives	Content	Periods
1	Define an algorithm	Concept of algorithm	**3**
2	Explain the advantage of a variable in an algorithm.	Variables	**3**
3	Explain operators to be used in an algorithm.	Expressions and Operators	**3**
4	Distinguish the functions from Reading (input) and those of displaying (output).	Reading and Writing expressions	**3**
5 -6	Give the situations of the use of various tests	Tests	**6**
7	Give the situation of the use of GOTO	Go to(goto)	**3**
8-9	Give the situations where we use various loops	Loops	**6**
10	Handle a table	Use a table in Algorithm	**3**
11	Revision		**3**
12	Exams		**3**
Total			**36**

2^{nd} term

Week	Specific objectives	Content	Periods
1-2	Define a program	Concept of program	**6**
3	Declare Variables and their type	Variables and their type	**3**
4	Declare Constants	The difference between a constant and a variable	**3**
5	Use of I/O functions	scanf() and printf()	**3**
6-7	Use operators	Arithmetic operators and their priorities	**6**
8-10	Use comparison and logic operators in conditional expressions	Comparison operators and their use with If, if...else, switch	**9**
		Logical operators and their use conditional	
		Priority of AND versus OR	
11	Revision		**3**
12	Exams		**3**
Total			**36**

3^{rd} term

Week	Specific objectives	Content	Periods
1-4	Use loops in c program	Loops	**12**
5-7	Create and Use functions	Function	**9**
8-10	Create and Use one dimensional array	One-dimensional array	**9**
11	Revision		**3**
12	Exams		**3**
Total			**36**

SENIOR FIVE

1. List of courses and chapters

Course: Introduction to computer II (60 periods)
Chapter1: Operating systems (20 periods)
Chapter 2: Using Linux Operating System (14 periods**)**
Chapter 3: Programming languages (6 periods)
Chapter 4: Introduction Networking (20 periods)

Course: Computer Programming (C++ language) (60 periods)
Chapter 1: Introduction to C++ programming (3 periods)
Chapter 2: Variable declaration (3 periods)
Chapter 3: I/O (2 periods)
Chapter 4: Operators (6 periods)
Chapter 5: Conditions and Loops (14 periods)
Chapter 6: Functions (12 periods)
Chapter 7: Arrays (7 periods)
Chapter 8: Structures (7 periods)
Chapter 9: Introduction to OOP (6 periods)

Course: Database (60 periods)
Chapter 1: Introduction to database (12 periods)
Chapter 2: Conceptual level (9 periods)
Chapter 3: Logical level (9 periods)
Chapter 4: Query language (30 periods)

2. General objectives

By the end of senior 5, Students should be able to:

- Analyse and suggest resolution to a given problem by illustrating succession of simple and accurate steps implicated in solving a given problem
- Draw a flowchart that shows steps implicated to solve a problem.
- Write a program in C++ language.
- Design a database
- Proficient in using most common operating systems

3. Course 1: Introduction to computers II (60 periods)

Chapter 1: Operating systems (20 periods)

- ❖ **Pre-requisites:** Introduction to computers I
- ❖ **Teaching aids**: Black/White board, chalk/marker, computer DOS or/and Windows
- ❖ **Skills to be developed**: Research, technical skills, communication skills
- ❖ **General objective**: Proficient in using DOS and WINDOWS

Specific objectives	Content	Suggested teaching and learning activities
State the evolution of OS	Definition of O.S History and evolution of O.S	The teacher boots a computer and explains why an O.S needs to be loaded.
Use basic DOS commands State and explain windows file systems	Use basic commands E.g.:DIR,MD,DEL,CD,EXIT Windows memory management system	The teacher demonstrates and helps learners to create a file in DOS, to open that file and change its directory

Chapter 2 Using Linux Operating System (14 periods)

- **Prerequisites**: DOS and Windows operating systems
- **Teaching aids**: White/Blackboard, marker /chalk, computer
- **Skills to be developed**: Thinking , technical
- **General objective**: Use Linux operating system

Specific objectives	Content	Suggested teaching and learning activities
Use the Linux desktop	Start a program Linux shell Basic Linux commands (ls, cd, whois, man, exit) File management Search a file	The teacher starts a computer with Linux operating system The teacher describes the desktop of Linux O.S and helps learners to discover the difference between Windows O.S and Linux O.S The learners interact with the computer using some basic commands

Chapter 3. Programming languages (6 periods)

- **Pre-requisites**: Programming language
- **Teaching aids**: White/Blackboard, marker /chalk, computer
- **Skills to be developed: thinking**: Technical skills
- **General objective**: To state all the generations of programming languages

Specific objectives	Content	Suggested teaching and learning activities
Explain what a machine language is	Machine language Assembly language High level language Fourth generation	The teacher gives an advantage of modern programming languages over primitive languages

Chapter 4. Introduction to networking (20 periods)

- **Pre-requisites**: Basic computing, Operating system
- **Teaching aids**: White/Blackboard, marker pen /chalk, computer
- **Skills to be developed**: Thinking , technical skills
- **General objective**: To explain basic concepts of network services

Specific objectives	Content	Suggested teaching and learning activities
Explain parts of an IP address	Types of networks 2 parts of an IP address	The teacher lists a set of IP addresses , shows the parts of each IP address and gives exercises of identifying the parts of IP addresses
Explain the use of DNS	Relationship between IP and DNS	The teacher explains the importance of each service.
Explain the use of common protocols	URL HTTP WWW FTP SMTP POP IMAP	

4. Course 2: Computer programming

Chapter 1: Introduction to C++ (3 periods)

- **Pre-requisites:** C programming language
- **Teaching aids:** Personal computer with Windows XP or new version of Windows, C++ compiler, black/white board, chalk/marker.
- **Skills to be developed**: Observation, analysis and research skills
- **General objective**: To identify the differences between C and C++
- **Methodological notes**:
  - This course of computer programming should focus on the other subjects within the combination (Mathematics, Physics and Economics).

Specific objectives	Content	Suggested teaching and learning activities
State the evolution of C++	The evolution of C++	The students finds the evolution of C++ programming language using internet or different books
Differentiate between C and C++	Difference between C and C++	The teacher writes one simple program in C and writes its equivalent in C++ The learners outline differences between the two written programs
Define the OOP	Particularities of C++	The learners outline what is particular to C++

Chapter 2.Variable declaration (3 periods)

- **Pre-requisites**: C programming
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: To declare variables of different types in C++

Specific objectives	Content	Suggested teaching and learning activities
State variable types	Types of variables	The teacher lists different values and shows that they are not of the same type. The teacher gives exercises for application
Declare and use a variable	Variable declaration	The teacher draws memory and shows how much space is occupied by variables of different types.
Declare and use a constant	Constant declaration	The teacher declares variables using const and #define keywords. The teacher gives exercises for application

Chapter 3.Input/ output (2 periods)

- **Pre-requisites**: C programming
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: To use input and output in C++

Specific objectives	Content	Suggested teaching and learning activities
Use input/output functions	The iostream.h library function <<cin>>and <<cout>>	The teacher writes a program to enable input and output. The teacher gives exercises for application

Chapter 4.Operators (6 periods)

- **Pre-requisites**: C programming
- **Teaching aids:** Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed:** Technical, thinking skills
- **General objective**: To use arithmetic, logical and comparison operators in C++

Specific objectives	Content	Suggested teaching and learning activities
Use arithmetic operators	Binary operators Unary operators	The teacher writes programs using binary and unary operators. The teacher gives exercises for application
Use comparison operators	If, if…else and switch	The teacher writes programs with conditions and gives exercises for application.
Use logical operators	Logical operators(AND,OR,NOT)	The teacher writes program using AND,OR and NOT operators The teacher gives exercises for application

Chapter 5. Conditions and Loops (14 periods)

- **Pre-requisites**: C programming
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: To use repetition statements in C++

Specific objectives	Content	Suggested teaching and learning activities
Use conditional expressions with comparison operators	**Comparison operators** “<”, “>”, “<=”, “>=”, “==”,”! = " and their use with If, if...else, switch	The teacher draws flowcharts to illustrate each program The teacher writes programs to compare two or more values and gives exercises for application.
Use a for loop	For loop	The teacher draws a flowchart writes a program with loop repetition. The teacher gives exercises for application
Use a while loop	While loop	The teacher draws a while flowchart and writes programs with while loop. The teacher gives exercises for application.
Use a do…while loop	Do…while	The teacher draws a do…while flowchart and writes programs with do…while loop. The teacher gives exercises for application.

Chapter 6.Function (12 periods)

- **Pre-requisites:** C programming functions
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: To create and use functions in C++

Specific objectives	Content	Suggested teaching and learning activities
Declare a function	Define a function Specify a function	The teacher writes a statement to declare a function and gives exercises for application.
Use variables inside and outside functions.	Local variables Global variables Static variables	The teacher writes a function having its variable declared inside of its body The teacher writes a function having its variable declared outside

	Dynamic variables	of its body The teacher gives exercises for application.
Use functions with parameters	Functions with one parameter Functions with two or more parameters	The teacher shows the difference in syntax between the declaring, the calling and defining statement of a function with parameter The learners point out the difference between the declaring, the calling and defining statement of a function without parameter The teacher gives exercises for application.

Chapter 7. Arrays (7 periods)

- **Pre-requisites:** One dimensional array in C programming
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: To use arrays in C++

Specific objectives	Content	Suggested teaching and learning activities
Declare and use an array of two dimensions of values from given data types	Array declaration use an array	The teacher draws an array of two dimensions indicating where values are stored Exercise on how the values are stored in the array

Chapter 8. Structures (7 periods)

- **Pre-requisites**: Arrays in C and C++ programming
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: Student should be able to create and use structures in C and in C++

Specific objectives	Content	Suggested teaching and learning activities
Define a structure State differences between a structure and arrays Declare and use a structure	Definition of structures Difference between an array and structure Structure declaration Declaration and Using Structure variable	The teacher shows the memory space occupied by a structure variable The teacher writes a program to access a variable inside a structure. Exercises of application

Chapter 9.Introduction to OOP (6)

- **Pre-requisites:** Structures in C and C++ programming
- **Teaching aids**: Black/White board, chalk/marker, computer with a C++ compiler
- **Skills to be developed**: Technical, thinking skills
- **General objective**: To define OOP concepts

Specific objectives	Content	Suggested teaching and learning activities
Define common concepts of OOP	Class Object Encapsulation Polymorphism Inheritance Characteristics and advantages of OOP	The teacher gives examples of natural objects and show that each object is of a specific class The teacher shows that a class may have a parent (base) and child (derived) classes Exercises of application

5. Course 3: Database

- **Pre-requisites:** Introduction to computers I & II
- **Teaching aids**: Computer with DBMS, black/white board, chalk/marker, projector
- **Skills to be developed**: Research, social, communication and technical skills
- **General objective**: Using computer in data management

Specific objectives	Content	Suggested teaching and learning activities
Define a database Give the importance of database	**Introduction to database** • Definition • Example (in Rwandan society) • Data independence • Data access • Data integrity and safety • Data recovery after a breakdown • Concurrent Transaction	The teacher describes and gives all kind of information flowing in companies (Banks, schools, etc) In groups students discuss and present the traditional ways of storing information Teacher helps students to compare the traditional ways of information storage on the computer model.
To explain the level or the steps to make a good database	**Conceptual level** Entity, Association, Property (Attribute, Identifier, Occurrence, Cardinality) • Weak entities • Hierarchy of the entities • Role **Logic level**: **Relations** • Two parts of a relation: Instance which is a table	Using an example, the teacher explains an entity as a category, a class of objects, individuals, etc. From a given number of occurrences, the teacher explains how to choose the most qualified property to be IDENTIFIER by using students list Exercises in computers laboratory

Use the interrogation languages theory to formulate relational algebra	having columns (cardinality) and fields (Degree) • Diagram to specify the name of columns, the field names and each type • Constraint of integrity • Primary key • Foreign key • A view **Any Query language** • Projection • Selection • The difference • Union • The cartesian product • Intersection • Division • Join	Exercises on formula drafting in computers laboratory

6. Lesson Distribution per term (12 weeks per term)

Course: Introduction to computer II

Term 1

Week	Specific objectives	Content	Periods
1-2	Definition of O.S	Definition of O.S	**4**
3-5	State the evolution of OS	History and evolution of O.S	**6**
6-8	Use basic DOS commands	Use basic commands	**6**
9-10	State and explain windows file system	Windows memory management system	**4**
	Revision		**2**
	Exams		**2**
Total			**24**

Term 2

Week	Specific objectives	Content	Periods
1- 8	Use the Linux desktop	Start a program	**2**
		Linux shell	**4**
		Basic Linux commands	**6**
		File management and Search file	**4**
9- 10	Explain what a machine language is	Machine language Assembly language	**2**
		High level language Fourth generation	**2**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term 3

Week	Specific objectives	Content	Periods
1-3	Explain parts of an IP address	Types of networks 2 parts of an IP address	**6**
4-5	Explain the use of DNS	Relationship between IP and DNS	**4**
6-10	Explain the use of common protocols	Protocols	**10**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Course: Computer programming

Term 1

Week	Specific objectives	Content	Periods
1	State the evolution of C++	The evolution of C++	**2**
2	Differentiate between C and C++	Difference between C and C++	**2**
3	Define the OOP	Particularities of C++	**2**
4	State variable types	Types of variables	**2**
5	Declare and use a variable	Variable declaration	**2**
6	Declare and use a constant	Constant declaration	**2**
7-8	Use input/output functions	The iostream.h library function <<cin>>and <<cout>>	**4**
9-10	Use arithmetic operators	Binary operators Unary operators	**4**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term2

Week	Specific objectives	Content	Periods
1-2	Use comparison operators	If, if…else and switch	**4**
3	Use logical operators	Logical operators	**2**
4-5	Use of a for loop Use of a while loop Use of a do…while loop	Loop	**4**
6	Declare a function	Define and specify a function	**2**
7-9	Use variables inside and outside functions	Variables	**4**
10	Use functions with parameters	Functions with one parameter Functions with two or more parameters	**4**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term 3

Week	Specific objectives	Content	Periods
1-3	Declare and use an array of two dimensions of values from given data types	Array	**6**
4-7	Define a structure State differences between a structure and arrays Declare and use a structure	Definition of structures Difference between an array and structure Structure declaration Declaration and using structure variable	**8**
8	Define common concepts of OOP	Class, Object	**2**
9		Encapsulation Polymorphism	**2**
10		Inheritance Characteristics and advantages of OOP	**2**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Course: Database

Term 1

Week	**Specific objectives**	**Contents**	**Periods**
1-2	Define a database	Definition and Examples	**6**
3-4	To give the importance of the data bases	Data independence, access, integrity and safety, Recovery after a breakdown, concurrent transactions.	**6**
5-7	To explain the level or the steps to make a good data bases	Conceptual Level	**9**
8-10		Role , Logical Level	**9**
11	Revision		**3**
12	Exams		**3**
Total			**36**

Term 2

Week	**Specific objectives**	**Contents**	**Periods**
1-8	To explain the level or the steps to make a good data bases	Two parts of a relation:	6
		Diagram to specify the name of column, the field names and each type.	6
		Constraint of integrity, Primary key, Foreign key.	6
		A view	6
9-10	Use the interrogation languages theory : to formulate relational algebra	Projection	6
11	Revision		3
12	Exams		3
Total			**36**

Term 3

Week	**Specific objectives**	**Contents**	**Periods**
1 -10	Use the interrogation languages theory : to formulate relational algebra	Selection, the difference, union	10
		The cartesian product, intersection	10
		Division, join	10
11	Revision		3
12	Exams		3
Total			**36**

SENIOR SIX

1. List of courses and chapters

Course: Computer Maintenance (60 periods)
Chapter 1: Computer Hardware (7 periods)
Chapter 2: Computer Software (9 periods)
Chapter 3: Computer Security (4 periods)
Chapter 4: Computer Network (40 periods)

Course: Database (60 periods)
Chapter 1: Structured Query language (20 periods)
Chapter 2: Database management system (**DBMS)** (20 periods)
Chapter 3: Database security (20 periods)

Course: Programming (90 periods)
Chapter 1: Class, Object and Encapsulation (30 periods)
Chapter 2: Introduction to event oriented programming (17 periods)
Chapter 3: Variable, Test and Loops in Visual Basic (23 periods)
Chapter 4: Function and List control in Visual basic (20 periods

2. General objectives

By the end of senior 6, the learner should be able to:

- Assemble, disassemble, prevent a computer and build a peer to peer network
- Maintain and secure a database
- Write a program in Visual Basic

3. Course 1: Computer Maintenance

Chapter 1: Computer Hardware (7 periods)

- **Pre-requisites:** Introduction to computer
- **Teaching aids:** Personal computer, Black board, Chalk
- **Skills to be developed:** Practical skills
- **General objective:** Identify computer physical component, assemble and disassemble a computer

Specific objectives	Content	Suggested teaching and learning activities
Connect a computer	➢ Identify the connectors for: ▪ Keyboard ▪ Mouse ▪ Screens (VGA, DVI) ▪ Printer (USB, parallel ,serial) ▪ Loudspeaker ➢ Electrically connect a computer with or without UPS ➢ Power necessary for the UPS ➢ What the inverter makes it possible to protect ➢ Power necessary for a regulator ➢ What the regulator makes it possible to protect ➢ The power which delivers an electrical connector is limited ➢ AT, ATX, BTX format	Learners connect computer to make it fully functional
Identify mother Board elements	➢ Identification and role of the following elements: ▪ chipset northern and southern ▪ Crush system BIOS ▪ CPU	Learners list the elements found on a specific mother board.

Assemble and disassemble a computer	▪ jumpers, switches ▪ Connector: PS/2, port series, Parallel port, USB, VGA ▪ IDE ▪ SIMM, DIMM ▪ Power supply connector ▪ CPU socket (CPU slot) ➢ ESD (Electrostatic discharge) effects. ➢ The existence of protection material ESD ➢ Set the mother board inside the case ▪ Set the hard disks and CD /DVD drive in the case. ▪ Insert the extension cards ▪ Connect the power supplier to the motherboard ▪ Insert the processor and jumpers configuration ▪ Set up of RAM memory ▪ Connect the panel	Teacher helps students to mount and dismount a computer

Chapter 2: Computer Software (9 periods)

- **Pre-requisites:** Introduction to computer
- **Teaching aids:** Personal computer, Black board, Chalk
- **Skills to be developed:** Practical skills
- **General objective:** Install an operating system and any application software

Specific objectives	Content	Suggested teaching and learning activities
Install an Operating System and application software	➢ How to launch the installation ➢ Boot from a CD ➢ Partitioning and formatting ➢ Installation of Windows XP, Linux ➢ Installation of computer drivers ➢ Multi-boot ➢ Installation of application software(MS office) ➢ Software Licences ➢ System Requirement to install and use a software (RAM, Disc, CPU)	Exercise on how to install the operating system and application software (MS office Package)
Configure the computer by using the possibilities offered by the control panel	➢ System icon ➢ Peripherals configuration ➢ IRQ, DMA, I/O main operation ➢ Configuration of: ▪ modem ▪ network ▪ printer ▪ scanner ▪ video and sound cards ▪ fire wall	Teacher helps students to configure computer peripherals via the windows control panel

	➢ add/delete programs	

Chapter 3: Computer Security (4 periods)

- ❖ **Pre-requisites**: Introduction to computer
- ❖ **Teaching aids:** Personal computer, Black board, Chalk, Anti-Virus
- ❖ **Skills to be developed:** Practical Skill
- ❖ **General objective:** Protect a computer

Specific objectives	**Content**	**Suggested teaching and learning activities**
The learner should be able to: Protect a computer	➢ Define a virus ➢ Update an antivirus ➢ Repair files, remove virus from computer, put in quarantine virus ➢ Danger of an infected flash disk when you are starting a Computer ➢ Means of transmission (media, network, Internet, email, removable media…) ➢ The type of damage create by the viruses Adware and Spyware ➢ Antivirus ➢ Anti spyware ➢ Fire wall ➢ Concept of encoding password (OS, BIOS)	The teacher demonstrates how to run an anti virus, to detect, to remove and to put a virus in quarantine Exercises of application
Use various methods of maintenance	Preventive maintenance: ➢ Regular cleaning ➢ Properly shut down the computer ➢ Use of maintenance tools (Scandisk, Defragmentation …)	The teacher helps students to clean outside and inside the computer. Students launch scandisk, defragmentation tools.

Chapter 4: Computer network (40 periods)

- **Pre-requisites:** Introduction to computer
- **Teaching aids:** Personal computer, Black board, Chalk ,UTP cable, Cable tester, RJ 45 connectors, Switch/Hub, Access point
- **Skills to be developed**: Practical skills
- **General objective:** Build a Peer to Peer network

Specific objectives	Content	Suggested teaching and learning activities
Transmission median	Transmission median ➢ Guided median UTP/STP cable Coaxial cable	The teacher helps students to make cross and straight network cables
Make a cable UTP/STP	▪ Fibre optic cable ➢ Unguided ▪ Infrared ▪ Bluetooth ▪ Wireless ➢ Network devices ▪ Switch and hub ▪ Access point ▪ Personal computer ➢ Network median ▪ Cable	
Identify basic peer to peer network components	▪ wireless ➢ Network peripheral	The students describe all devices that participate in Peer-to-Peer network

	▪ Printer ▪ Scanner ▪ Hard disk... ➢ What a modem can do ➢ Internal modems and external modems ➢ Modify the configuration ➢ Install a NIC Card ➢ Make a cross over cable ➢ Make a straight cable ➢ Connect computers to a Switch/Hub ➢ Configure IP addresses ➢ File sharing ➢ Printer sharing	
Identify and configure a modem	Modem installation Modem configuration	The teacher helps students to install and to configure a modem
Build a peer to peer network	Peer to peer network	The students build a peer to peer network

4. Course 2: Database

Chapter 1: Structured Query Language (20 periods)

- **Pre-requisites:** Relation algebra and Relation Calculus
- **Teaching aids:** Black/Whiteboard, chalk
- **Skills to be developed:** Problem resolving skills and Practical skills
- **General objective:** Use queries

Specific objectives	Content	Suggested teaching and learning activities
Use the QBE(Queries by example)	➢ And/or queries ➢ Junction ➢ Aggregation (avg, count, min max sum.) ➢ Tuples inserting. ➢ To remove and validate	Exercise on aggregations

Chapter 2: Database management system (20 periods)

- **Pre-requisites:** Structured queries language
- **Teaching aids:** Black/Whiteboard, chalk, DBMS software
- **Skills to be developed:** Practical skills
- **General objective: U**se queries

Specific objectives	Content	Suggested teaching and learning activities
Use SQL on the data bases	➢ To create a table ➢ To add and remove tuples ➢ To program a primary key ➢ A condition (where) ➢ Aggregation ➢ A foreign Key in SQL ➢ To reinforce the integrity ➢ Creation of views ➢ Query on a table ➢ Query on multiple tables ➢ Nested queries	Exercises on how to formulate basic sql expression

Chapter 3: Database security (20 periods)

- **Pre-requisites**: Database management system
- **Teaching aids**: Black board, chalk, DBMS Software
- **Skills to be developed**: Practical skills
- **General objective**: to implement security on the database

Specific objectives	Content	Suggested teaching and learning activities
Secure a database	➢ Integrity ➢ The Privacy ➢ The availability ➢ Command GRANT and REVOKES ➢ Backup	Exercises on how to formulate safety or security using SQL Queries
Manage and administrate a data base	➢ Concept of transaction ➢ Concurrent access ➢ Remote access	

5. Course 3: Programming Language

Chapter 1: Class, Object and Encapsulation (30 periods)

- **Pre-requisites**: Introduction to Objected Oriented Language
- **Teaching aids**: Black/Whiteboard, chalk, Computer with Windows XP, Visual C++
- **Skills to be developed**: Practical skills and research skills
- **General objective**: Use class and Explain encapsulation

Specific objectives	Content	Suggested teaching and learning activities
Use classes and objects encapsulation	➢ Class creation ➢ Object creation ➢ Encapsulation	To describe the syntax of a class. Exercises on how to write program using Classes
Use constructors and destructors	➢ Function call ➢ Definition of the term ➢ Writing of constructions ➢ Writing of destructions	Exercises on how to rewrite old programs each one with a constructor and a destructor

Chapter 2: Introduction to event oriented programming (17 periods)

- **Pre-requisites:** Introduction to objected oriented
- **Teaching aids:** Black/Whiteboard, chalk, Computer with Windows XP, Visual studio
- **Skills to be developed:** Research skills and practical skill
- **General objective:** Compile and execute a program in Visual basic

Specific objectives	Content	Suggested teaching and learning activities
The learner should be able to:		
Define an events oriented language	➢ Introduction to the event oriented language ➢ Objects Concept ➢ Events Concept	Describe an event in data processing. Generate events of simple type Click, double click, right click …
Use various Controls in Visual BASIC	➢ Create a new project in VB ➢ Elements of the IDE ➢ Controls	Insert a form object in a project, save it and exit. Insert various controls on an Application Open projects in various ways

Open an existing project Describe the Properties, Syntax, Methods and Procedures of events. Compile a simple application in Visual BASIC	▪ Form ▪ Command buttons ▪ Labels ▪ Text Box ➢ Localization of the project ➢ launch the project ➢ Object Properties(Control) ➢ Syntax ➢ Methods and events ➢ Event-driven Procedures ➢ Compilation and execution in Visual Basic	launch visual BASIC, insert an objects and to identify the event-driven properties, Syntax, Methods and Procedures Launch an application and to compile it Visual Basic errors identification and correction

Chapter 3: Variable, Test and Loops in Visual Basic (23 periods)

- ❖ **Pre-requisites:** Introduction to event oriented programming
- ❖ **Teaching aids:** Black board, chalk, Computer with Windows XP, Visual studio
- ❖ **Skills to be developed:** Practical skill and research skill
- ❖ **General objective:** Use variable, test and loop in Visual basic program

Specific objectives	Contents	Suggested teaching and learning activities
Use the variables, Operators, Test and loops in Visual BASIC. Use the objects (Controls) which cause the utilisation of tests	➢ Declaration and variable types ➢ Operators ➢ Tests ➢ The objects Combo box, List, Options box and Check box. ➢ Loops (For… Next, Do… While)	Create an application proposed by the teacher or of their own choice under the supervision of their teacher

Chapter 4: Function and List control in Visual basic (20 periods)

- **Pre-requisites:** Introduction to event oriented programming
- **Teaching aids:** Black/Whiteboard, chalk, computer with Windows XP, Visual studio
- **Skills to be developed:** Practical skill and research skill
- **General objective:** Use function and list control in Visual basic program

Specific objectives	Contents	Suggested teaching and learning activities
Write the functions in Visual BASIC	➢ Input Box, Msgbox Functions. ➢ Functions sqr (), val (), str ()	Use inputBox (), msgBox (), val () functions in a simple VB program
Insert Lists Controls and to write the corresponding code	➢ ListBox ➢ ComboBox ➢ DriveListBox ➢ DirListBox ➢ FileListBox	Insert list control on an application in Visual BASIC
Create Menus	➢ Creation and insertion of the menus and submenus	Insert Menus and submenus on a Form
Write various graphic elements in Visual BASIC.	Elements of Graphs	Use some graphic elements on an application in Visual BASIC

6. Lesson Distribution per term

Course: Computer Maintenance
Term 1

Week	**Specific Objectives**	**Content**	**Periods**
1-3	Identify computer physical components Assemble and disassemble a computer	Computer Hardware	**7**
4- 8	Install an operating system and common application software	Computer Software	**9**
9-10	Protect a computer	Computer Security	**4**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term 2

Week	**Specific Objectives**	**Content**	**Periods**
1-3	Make a UTP/STP cable	Transmission media	**6**
4-10	Identify basic peer to peer network components	Network devices	**6**
		Network media	**4**
		Network peripherals	**4**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term 3

Week	**Specific Objectives**	**Content**	**Periods**
1-2	Identify and configure a modem	Modem configuration	4
3-10	Build a peer to peer network	Network set up	16
11	Revision		2
12	Exams		2
Total			**24**

Course: Database
Term 1

Week	Specific Objectives	Content	Periods
1-2	Use the QBE(Queries by example	And/Or queries Junction	**4**
3-4		Aggregation (avg, count, min max sum.)	**4**
5-6		Insert tuples To remove and validate	**4**
7-8	Use SQL on the data bases	To create a table	**4**
9-10		To add and remove tuples Specify a primary key	**4**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term 2

Week	Specific Objectives	Content	Periods
1-9	Use SQL on the data bases	➢ A condition (where) ➢ Aggregation ➢ A foreign Key in SQL	**5**
		➢ To reinforce the integrity ➢ Creation of views	**5**
		➢ Query on a table ➢ Query on multiple tables ➢ Nested queries	**8**
10	Secure a data base	➢ Integrity	**2**
11	Revision		**2**
12	Exams		**2**
Total			**24**

Term 3

Week	Specific objectives	Content	Periods
1- 5	Secure a data base	➢ The Privacy ➢ The availability	4
		➢ Command GRANT and REVOKES ➢ Backup	6
6-10	Manage and administrate a data base	➢ Concept of transaction ➢ Concurrent access ➢ Remote access	10
11	Revision		2
12	Exams		2
Total			24

Course: Programming

Term 1

Week	Specific objectives	Content	Periods
1-5	Use classes and objects Encapsulation	Class creation	4
		Object creation	4
		Function call	7
6-10	Use constructors and destructors	Definitions of the terms	4
		Writings of constructors	7
		Writings of destructors	4
11	Revision		3
12	Exams		3
Total			36

Term 2

Week	Specific objectives	Content	Periods
1-2	Define an events oriented language	Define an object	**2**
		Define an event	**2**
		Create a new project in VB	**2**
		Elements of VB IDE	**2**
3-5	Describe the properties, syntax, methods and procedures of events.	Objects properties(Control)	**2**
		Syntax	**2**
		Methods and events	**2**
		Event-driven Procedures	**2**
6	Compile a simple application in Visual Basic.	Compilation and execution in VB	**1**
6-10	Use the variable, Operators, Test and Loops in VB	The code elements	**2**
		Declaration and variable types	**2**
		Operators	**2**
		Tests	**4**
		The objects Combo box, List, Options box and Check box	**3**
11	Revision		**3**
12	Exam		**3**
Total			**36**

Term 3

Week	**Specific objectives**	**Content**	**Periods**
1-3	Use of Test and Loops in Visual Basic	Loops (for… Next, Do…while)	**4**
		Input Box, Msgbox Functions.	**2**
		Functions sqr(), val(), str()	**4**
4-7	Insert Lists Controls and write the corresponding code	ListBox	**2**
		ComboBox	**2**
		DriveListebox	**2**
		DirListBox	**2**
		FileListBox	**2**
8	Create Menus	Creation and insertion of the menus and submenus	**4**
9-10	Write various graphic element in Visual Basic	Colors	**1**
		Shape	**1**
		Line	**2**
		Frame	**2**
11	Revision		**3**
12	Exams		**3**
Total			**36**

VII. REFERENCES

1. Cisco academy documentations.1999
2. CLAUDE DELANNOY, *Programmation en C*, eyrolles
3. Collins Ritchie, *Modern Operating Systems*
4. DAVID I. SCHNEIDER, *Essentials of visual basic 6.0*
5. Lisa Ronald and James Chellis: Windows 2003 Server, Study Guide, 2002
6. NCDC: Professional courses syllabus for computer science section, 2006
7. PRINTICE HALL, *Programming*, Upper Saddle River, New Jersey 0 7458,1999
8. Raghu Ramakrishnan et Johannes Gehrke, *Database Management Systems second edition*, mac graw Hill.
9. ROBERT LAFORE, Objetct Oriented programming in Turbo C++, The waite Group ,Inc.Calfonia, 1991.
10. Robert Lafore, *Objetct Oriented programming in Turbo* C++, The waite Group Press, 1991.
11. Sisco: SISCO academy 2002 books
12. Waite Group: Turbo C, 1990
13. www.bced.gov.ca
14. www.computer.org
15. www.cprogramming.com
16. www.dmoz.org/Computers/Hardware/Peripherals
17. www.java2s.com
18. www.ltscotland.org.uk

VIII. APPENDIX: WEEKLY TIME ALLOCATION

a. Physics-Computer science -Mathematics option (PCM)

Subject	**Number of periods**		
	S4	**S5**	**S6**
Core subjects (all compulsory and examinable)			
Physics	7	7	7
Computer science	7	7	7
Mathematics	7	7	7
Entrepreneurship	7	7	7
General Paper	2	2	2
SUB-TOTAL	**30**	**30**	**30**
Non Examinable Subjects ***(Schools may choose from the list of non examinable subjects below but English and Kinyarwanda are Compulsory)***			
Technical drawing	2	2	2
Chemistry	2	2	2
Kinyarwanda	2	2	2
English	2	2	2
Co-curricular activities:			
Sport, culture activities, Clubs, religious studies, study, research in library…	2	2	2
TOTAL	**36**	**36**	**36**

b. Computer Science-Economics-Mathematics option (CEM)

Subject	Number of periods		
	S4	**S5**	**S6**
Core subjects (all compulsory and examinable)			
Computer science	7	7	7
Economics	7	7	7
Mathematics	7	7	7
Entrepreneurship	7	7	7
General Paper	2	2	2
SUB-TOTAL	**30**	**30**	**30**
Non Examinable Subjects ***(Schools may choose from the list of non examinable subjects below but English and Kinyarwanda are Compulsory)***			
Technical drawing	2	2	2
Kinyarwanda	2	2	2
English	2	2	2
Co-curricular activities			
Sport, culture activities, Clubs,religious studies, study, research in library…	2	2	2
TOTAL	**36**	**36**	**36**