MARIAN UNIVERSITY COLLEGE
(Constituent College of St. Augustine University of Tanzania)

FACULTY OF NATURAL AND APPLIED SCIENCES

BACHELOR OF SCIENCE IN COMPUTER SCIENCE
DEGREE PROGRAMME

MAY 2015
# Table of Content

1.0 INTRODUCTION .................................................................................................................. 3

1.1 NAME OF THE DEGREE PROGRAMME AND CAPACITY OF STUDENT ENROLLMENT .................................................................................................................. 3

2.0 PROGRAMME OBJECTIVES ............................................................................................... 4

3.0 SUMMARY AND SEMESTRISATION OF THE COURSES ............................................. 5

3.1 COURSE LIST .................................................................................................................... 5

3.2 DETAILED COURSE DESCRIPTIONS ............................................................................... 7

4.0 ACADEMIC REGULATIONS ............................................................................................ 45

4.1 TEACHING AND LEARNING FACILITIES AND RESOURCES ................................. 45

4.3 ACADEMIC STAFF FOR THE FIRST ACADEMIC YEAR 2015/2016 ......................... 45

4.4 ADMISSION REQUIREMENTS ......................................................................................... 46

4.5 COURSE COMPLETION AND GRADING SYSTEM ....................................................... 46

4.6 ADMISSIONS AND EXAMINATION REGULATIONS .................................................. 47

4.7 REGULATIONS FOR PROJECT/FIELD WORK ............................................................ 47

4.7.1 AIMS AND OBJECTIVES OF FIELD ATTACHMENT ........................................... 47

4.7.2 NATURE OF THE TRAINING .................................................................................... 48

4.7.3 FINANCIAL ASPECTS OF THE PROGRAMME ....................................................... 48

4.7.4 METHOD OF ASSESSMENT OF FIELD PRACTICAL TRAINING ....................... 48

5.0 APPENDICES ................................................................................................................... 49
1.0 INTRODUCTION
The role of Information and Communication Technology (ICT) in national development has recently been acknowledged worldwide. This is particularly so because ICT permeates all sectors of the economy including education, agriculture, governance and health. ICT is now considered as one of the pillars that support economic development and poverty reduction, as it enables information and knowledge to travel faster and further for making timely and informed decisions. The Tanzania development Vision 2025 and the Tanzania ICT policy of 2005 basically attest to the importance of Tanzania using ICT in facilitating economic development and governance issues. However, the development of ICT in Tanzania, as the case in other African countries, is faced with inadequate supply of human capital to carry out the development of ICT Sector.

In this regard, Marian University College (MARUCO) has adopted a Bachelor of Science in Computer Science degree programme from Ruaha Catholic University [formally Ruaha University College], in a bid to attain a well educated Tanzanians in the ICT and science based disciplines. The ICT programme is a timely response to develop a required human capital with requisite skills to manage the ICT sector in Tanzania. Thus MARUCO will offer a three-year course in Computer Science leading to the award of BSc in Computer Science.

Therefore, this document describes in detail the course structure of Computer Science including Programme objectives, Detail Course Descriptions, and Academic Regulations.

1.1 NAME OF THE DEGREE PROGRAMME AND CAPACITY OF STUDENT ENROLLMENT

The name of the degree programme will be Bachelor of Science in Computer Science (BSc. Comp. Sci.).

The three (3) years programme is aimed at producing professionals and technicians who will work in areas which apply computer skills in information communication technology. The projected student enrolment is given in the Table below:

<table>
<thead>
<tr>
<th>S/No.</th>
<th>Name of the Programme</th>
<th>Number of Students per programme</th>
<th>Total enrolment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bachelor of Science in Computer Science</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>
2.0 PROGRAMME OBJECTIVES

The programmes aim to produce

1. Graduates with knowledge and the ability to construct substantial programs for users.
2. Graduates who are able to analyse the information requirements of organisations, to design appropriate systems within which computers may play a part, and with longer experience to implement these systems.
3. Graduates who are able to control the storage and transmission of data, the operation of computer hardware and software. Current names for such people are system administrators and network administrators.
4. Graduates who can also go on to do postgraduate studies in either ICT and become trainers in their areas of specialisation.
### 3.0 SUMMARY AND SEMESTRISATION OF THE COURSES

#### 3.1 COURSE LIST

##### 1ST YEAR COURSES

**First Semester**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Uni</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 100</td>
<td>2</td>
<td>Introduction to Informatics</td>
</tr>
<tr>
<td>BCS 110</td>
<td>2</td>
<td>Microcomputer Applications</td>
</tr>
<tr>
<td>BCS 106</td>
<td>3</td>
<td>Calculus</td>
</tr>
<tr>
<td>BCS 101</td>
<td>2</td>
<td>Computer Architecture</td>
</tr>
<tr>
<td>BCS 104</td>
<td>3</td>
<td>Discrete Structures</td>
</tr>
<tr>
<td>BMS 110</td>
<td>2</td>
<td>Introduction to Business Management</td>
</tr>
<tr>
<td>BPH 113</td>
<td>2</td>
<td>Social Ethics I</td>
</tr>
<tr>
<td>BSS 110</td>
<td>3</td>
<td>Development Studies I</td>
</tr>
<tr>
<td>BLG 108</td>
<td>3</td>
<td>Communication Skills I</td>
</tr>
</tbody>
</table>

**Second Semester**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Uni</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSC 102</td>
<td>3</td>
<td>00 Programming</td>
</tr>
<tr>
<td>BCS 103</td>
<td>3</td>
<td>Algoriths &amp; Data Structures</td>
</tr>
<tr>
<td>BMS 222</td>
<td>2</td>
<td>Small Businesses &amp; Entrepreneurship</td>
</tr>
<tr>
<td>BCS 105</td>
<td>3</td>
<td>Linear Algebra</td>
</tr>
<tr>
<td>BPH 104</td>
<td>2</td>
<td>Social Ethics II</td>
</tr>
<tr>
<td>BSS 130</td>
<td>3</td>
<td>Development Studies II</td>
</tr>
<tr>
<td>BLG 128</td>
<td>3</td>
<td>Communication Skills</td>
</tr>
</tbody>
</table>

**FIRST PRACTICAL TRAINING**

BPT 199  2  First Year Practical Training (6 to 8 weeks)

##### 2ND YEAR CORE COURSES

**First Semester**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Uni</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 200</td>
<td>3</td>
<td>Network Design &amp; Administration I</td>
</tr>
<tr>
<td>BCS 203</td>
<td>3</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>BCS 207</td>
<td>2</td>
<td>Structure of Programming Languages</td>
</tr>
<tr>
<td>BCS 202</td>
<td>3</td>
<td>Operating Systems</td>
</tr>
<tr>
<td>Course Code</td>
<td>Credits</td>
<td>Course Name</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>BCS 215</td>
<td>2</td>
<td>Probability and Statistics</td>
</tr>
</tbody>
</table>

**Second Semester**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Credits</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 204</td>
<td>2</td>
<td>MIS</td>
</tr>
<tr>
<td>BCS 201</td>
<td>3</td>
<td>Database Design</td>
</tr>
<tr>
<td>BCS 208</td>
<td>2</td>
<td>Computer Graphics and Multimedia</td>
</tr>
<tr>
<td>BCS 206</td>
<td>2</td>
<td>PC Diagnostics &amp; Maintenance</td>
</tr>
<tr>
<td>BCS 205</td>
<td>2</td>
<td>Linux System Administration</td>
</tr>
</tbody>
</table>

**SECOND PRACTICAL TRAINING**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Credits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPT 299</td>
<td>2</td>
<td>Second Year Practical Training (6 to 8 weeks)</td>
</tr>
</tbody>
</table>

**2ND YEAR OPTIONAL COURSES**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Credits</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 209</td>
<td>3</td>
<td>Advanced Software Engineering</td>
</tr>
<tr>
<td>BCS 210</td>
<td>3</td>
<td>Programming in C</td>
</tr>
<tr>
<td>BCS 211</td>
<td>2</td>
<td>GIS</td>
</tr>
<tr>
<td>BCS 213</td>
<td>2</td>
<td>Compilers</td>
</tr>
<tr>
<td>BCS 214</td>
<td>2</td>
<td>Theory of Computation</td>
</tr>
<tr>
<td>BCS 216</td>
<td>2</td>
<td>Numerical Analysis</td>
</tr>
</tbody>
</table>

**3RD YEAR CORE COURSES**

**First Semester**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Credits</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 305</td>
<td>2</td>
<td>Computer System Security</td>
</tr>
<tr>
<td>BCS 300</td>
<td>3</td>
<td>Systems Analysis and Design</td>
</tr>
<tr>
<td>BCS 304</td>
<td>2</td>
<td>Professional Practices of Information</td>
</tr>
<tr>
<td>BCS 399</td>
<td>4</td>
<td>Final ICT Project</td>
</tr>
</tbody>
</table>

**Second Semester**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Credits</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 315</td>
<td>2</td>
<td>Network Design &amp; Administration II</td>
</tr>
<tr>
<td>BCS 302</td>
<td>3</td>
<td>Database Systems</td>
</tr>
<tr>
<td>BCS 301</td>
<td>2</td>
<td>Internet Programming and E-</td>
</tr>
</tbody>
</table>

**3RD YEAR OPTIONAL COURSES**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Credits</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS 303</td>
<td>2</td>
<td>Project Management</td>
</tr>
<tr>
<td>BCS 310</td>
<td>2</td>
<td>Computer Simulation and Modelling</td>
</tr>
<tr>
<td>BCS 311</td>
<td>2</td>
<td>Introduction Artificial Intelligence</td>
</tr>
<tr>
<td>BCS 307</td>
<td>2</td>
<td>Operations Research</td>
</tr>
<tr>
<td>BCS 313</td>
<td>2</td>
<td>Distributed Systems</td>
</tr>
</tbody>
</table>
3.2 DETAILED COURSE DESCRIPTIONS

BCS 110 Microcomputer Applications 2 units

Course objectives
To provide a broad perspective of Microcomputers as a tool for processing. Students are expected to gain interaction with micro-computer system, know how the applications programs are taken care of by the OS, to summarise common network structures, have experience with DOS and Windows 95 during the course development.

Learning outcomes
At the end of the module, the students should:
1. Describe the components making a microcomputer and to describe how each works
2. Use famous microcomputer applications (wordprocessors, spreadsheets, database, etc)
3. Communicate with others via Internet email
4. Use the Internet to acquire information

Module Content
Microcomputer organisation. CPU, Input/Output units. Memory (access, location, address, contents, writing, reading). Arithmetic and logic units. Control unit (machine instruction, instruction counter). Machine cycle (fetch and execute). Internal representation of data (binary system, bits and bytes, octal and hexadecimal systems). (20 per cent.)

Computer architecture. Software & hardware, application software, system software, classification of computers (1st, 2nd, 3rd, 4th generations, etc supercomputers, mainframes, servers, workstations, minicomputers, microcomputers), (15 percent)

Computer networks. LAN, WAN, MAN, Client server, peer-to-peer, (15 percent)

Operating systems. History, DOS, MS-DOS, network operating systems, UNIX/LINUX, WINDOWS (GUI, icon, click, double click, files, folders, etc) (20 percent)

Microcomputer applications. Office applications (word, excel, Access, powerpoint, etc), Internet, email (25 percent)

Lab work. Students should be given small problems as homework. The lab sessions should be used to practice computer applications (25 percent).

Text Book
Essentials of computing by BCS.L. Capron

Reference:
Microcomputer applications by Robert Coarer and Pal K. Surgure
Computers in your future by Marilya Meyer and Roberta Baber
BCS 100 INTRODUCTION INFORMATICS 2 (2L, 1T)

UNITs Course objectives

To provide a broad perspective of Informatics and Microcomputers as a tool for processing.

Learning outcomes

At the end of the course, the students should be able to

1. Describe the structure and working of computers
2. Demonstrate skills for algorithm development and problem-solving

Module

Content


Data representation. Binary numbers, hexadecimal, octal, complement. Binary codes (ASCII, EBCDIC, etc.). Error detecting and error correcting codes. Multiplexers, encoders, decoders. Arithmetic circuits, number systems (integer, floating point). (10 percent)

Sets, relations, functions: Sets and (partially) ordered sets; subsets, partitions, coverings; permutations, combinations, elementary probability, counting techniques.

Relations: properties (symmetry, transitivity, functionality, totality,…), operations on relations - relational algebra.

Functions, function spaces; Scott continuity, fixed points. (30 per cent).

Inductive Model Building. Measuring the World, Summarizing data, Centrality and Dispersion, Fitting Data with regression

Algorithm development. Techniques of problem solving. Flowcharting. Stepwise refinement. Simple numerical examples. Algorithms for searching (e.g. linear, binary), sorting (e.g. exchange, insertion), merging of ordered lists. Examples taken from business application (compound/simple interest),. (25 per cent.)

Lab work. Students should be given small problems as homework. The lab sessions should be used practice design, code, debug and document short programs using a rational programming style. (25 percent).

Text Book

Essentials of computing by BCS.L. Capron

Reference:

Microcomputer applications by Robert Coarier and Pal K. Surgure
Computers in your future by Marilya Meyer and Roberta Baber
BCS 101 COMPUTER ARCHITECTURE 3 (2L, 1P)

UNIT Objectives
To understand the fundamentals of logic design; to understand the mechanics of data representation, data processing and control within a digital computer; to understand computer architecture (organisation and structures of the major hardware components of computers).

Learning outcomes
At the end of the course the student should be able to:
1. Describe the fundamentals of logic design
2. Show the mechanics of data representation, data processing and control within a digital computer
3. Describe the architecture of computers including organisation of the major hardware components
4. Describe the different types of machine instructions

Course Contents

<table>
<thead>
<tr>
<th>Part</th>
<th>Title</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part 1</td>
<td>introduction to the course.</td>
<td>7 %</td>
</tr>
<tr>
<td>1.</td>
<td>Structured Computer Organization</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Milestones in Computer Architecture</td>
<td></td>
</tr>
<tr>
<td>Part 2</td>
<td>MIPS Assembly language.</td>
<td>7 %</td>
</tr>
<tr>
<td>1.</td>
<td>The Von Neumann machine</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Instructions</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Registers,</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>Arithmetic, add; add translating high level arithmetic into assembly.</td>
<td></td>
</tr>
<tr>
<td>Part 3</td>
<td>More on Mips Assembly</td>
<td>7 %</td>
</tr>
<tr>
<td>1.</td>
<td>Memory and Registers, lw and sw. Addressing modes.</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Arrays. Some example programs with arrays.</td>
<td></td>
</tr>
<tr>
<td>Part 4</td>
<td>More on Mips Assembly</td>
<td>7 %</td>
</tr>
<tr>
<td>1.</td>
<td>Subroutines</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Stacks and the answer to recursive subroutines</td>
<td></td>
</tr>
<tr>
<td>Part 5</td>
<td>Revision of Digital logic.</td>
<td>7 %</td>
</tr>
<tr>
<td>1.</td>
<td>Logic gates, truth tables, implementing truth tables.</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Longest path.</td>
<td></td>
</tr>
<tr>
<td>Part 6</td>
<td>Arithmetic</td>
<td>5 %</td>
</tr>
<tr>
<td>1.</td>
<td>Binary numbers hex-numbers.</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>The concept of representation.</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Negative Numbers and Two’s complement</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>The evils of floating point.</td>
<td></td>
</tr>
<tr>
<td>Part 7</td>
<td>Implementing Addition, Ripple adder, Faster Adder.</td>
<td>4 %</td>
</tr>
<tr>
<td>Part 8</td>
<td>Implementing the MIPS processor</td>
<td>7 %</td>
</tr>
<tr>
<td>1.</td>
<td>Latches and memory</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>The single cycle approach.</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Problems with the single cycle approach. Slowest instruction gives the cycle time; a functional unit can only be used for one thing at a time.</td>
<td></td>
</tr>
<tr>
<td>Part 9</td>
<td>Multicycle implementations</td>
<td>8 %</td>
</tr>
<tr>
<td>1.</td>
<td>The concept of a cycle.</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Finite state machines.</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Balancing the work into single cycles.</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>Controlling the work done with a finite state machine.</td>
<td></td>
</tr>
</tbody>
</table>
5. The five cycles of the MIPS.

**Part 10. Implementing Finite State machines** [10 %]
1. Roms, PLAs
2. Microcode
3. RISC/CISC
4. Microcode and some modern processors.

**Part 11. Pipelines** [9 %]
1. Doing more than one thing at once.
2. Problems with pipelines, stalls, branch delay slots.
3. Making programs faster by avoiding stalls.

**Part 12. Caches.** [9 %]
1. Principle of locality
2. Direct Mapped Caches
3. Set Associative Caches
4. LRU
5. write through, write back
6. Cache line, length

**Part 13. Virtual Memory.** [6 %]

**Part 14. I/O Polling and Interrupts** [8 %]
1. Memory Mapped I/O
2. Interrupts, Polled I/O.

**Lab work.** Study of an actual, simple microprocessor (e.g. Motorola 68000). (20 percent)

**Readings**

   Patterson and Hennessy, Morgan Kaufmann/Elsevier

**BCS 102: OBJECT ORIENTED PROGRAMMING I 3 (2L, 1P) Units**
Prerequisite: BCS 100

**Objectives**

The course first aims to introduce to the students the object oriented (OO) programming paradigm and then to teach them how to write structured and object-oriented programs in Java. This will highlight the use of control structures, the use of language Libraries and other tools.

**Learning outcomes:**

At the end of the module, the student
1. should have a working knowledge of a widely used object oriented programming language
2. be able to design, code, debug and document object oriented programs.

**Course Content**
\textbf{Introduction.} Short Java history, program structure, Edit-Compile-Run cycle, Object Oriented Concepts, data types, variable declaration, constants, assignment statements, operators, string manipulation (24%)

\textbf{Input & Output.} Standard output & Input, other example of standard classes e.g. Math class, Gregorian calendar class (5%)

\textbf{Defining your own classes.} Instantiable class, constructors, visibility modifiers, local variables, return values, parameter passing, accessors, mutators, organizing classes into packages, making an instantiable class the Main class. (20%)

\textbf{Control Structures.} Algorithms, Stepwise refinements, Decisions (if, if-else, switch), Repetitions (for loop, while loop, do..while loop), The Break statement and the continue statement (13%)

\textbf{Arrays.} Array basics, multidimensional arrays, arrays of objects, passing arrays to methods (9%)

\textbf{Inheritance and Polymorphism.} Inheritance, Method overloading and overriding, Polymorphism, more modifiers: protected & default, static & final, static methods & variables, abstract classes and interfaces (13%)

\textbf{Exception handling.} Catching exceptions, throwing exceptions, propagating exceptions, types of exceptions, programmer-defined exceptions. (7%)

\textbf{File Manipulation.} File concepts, uses of files, text and binary files, sequential and random file access, Processing a text file, Random Access Files. (9%)

\textbf{LAB WORK:} The lab sessions should be used to test student programs (20 percent).

\textbf{READINGS}
1. Mark & Allen & Weiss, Data Structures and Problem solving using Java
3. Glenn BCS. Rowe, An Introduction to Data Structures and Algorithms with Java

\textbf{BCS 105 LINEAR ALGEBRA 3 (3L, 1T)}

\textbf{UNITS}

\textbf{OBJECTIVE}
To give students an introduction to certain important techniques of applied linear algebra with some understanding of the theories behind them.

\textbf{Learning outcomes:}
At the end of the course the students should be able to:
1. Define what are linear equations and vector spaces
2. Show the connection between linear equations and vector spaces
3. Solve linear equations either by Gauss elimination or by determinants
4. perform linear transformations on matrices or rigid bodies

\textbf{MODULE CONTENT}
\textbf{Linear equations and Matrices.} Systems of linear equations, matrices and matrix operations, algebraic properties of matrices, special types of matrices, inverses of matrices, equivalent matrices (30%)

\textbf{Real Vector Spaces.} Vectors in \(\mathbb{R}^2\) and \(\mathbb{R}^3\), vector spaces and subspaces, linear independence, bases and dimensions, coordinate and isomorphism, the standard inner product on \(\mathbb{R}^2\) and \(\mathbb{R}^3\), cross product, the Gram-Schmidt process (30%)
**Linear Transformations and Matrices.** The kernel and range of a linear transformation, the matrix of a linear transformation, the vector space of matrices and the vector space of a linear transformation (20%)  
**Determinants.** Properties of determinants, cofactor expansion, inverse of a matrix, applications of determinants (10%)  
**Eigen values and Eigen vectors.** Diagonalisation, diagonalisation of symmetric matrices (10%)

**BCS 106 CALCULUS 3 (3L, 1T) UNITS**

**OBJECTIVES**
To give students a formal study of the foundations of mathematics which have been approached very intuitively and to introduce them to the limit concept of analysis

**Learning outcomes:**
At the end of the course the student should be able to:
1. Describe the concept of function and its domain and range
2. Calculate the limit of function at a given point
3. Perform both analytical and numerical differentiation and integration of functions of one or more variables.

**MODULE CONTENT**

**Functions and Graphs.** Coordinates and lines, circles and graphs of equations, functions and graphs of functions (10%)  
**Differential calculus.** Limits and continuity, the derivative and differentiation, Extreme function values, techniques of graphing (40%)  
**Integral calculus.** The definite integral and integration, inverse functions, logarithmic and exponential functions, inverse trigonometric functions, techniques of integration, indeterminate forms and Taylor’s formula, sequences and infinite series of constant terms, Power series (50%)  

**Text/reference books:**
Karplan, Wilfred & Lewis, D. J, *Calculus and Linear*, John Wiley & Sons  
Apostol, T, *Calculus Volume II*, John Wiley & Sons  

**BCS 215 PROBABILITY AND STATISTICS 2 (2L, 1T)**

**UNITS OBJECTIVES**
To develop an understanding of the basic concepts of statistics and probability with emphasis on practical applications.

**Learning outcomes:**
At the end of the course the student should be able to:
1. Use the basic concepts of statistics and probability in the solution of related problems
2. Use the most commonly used terminologies in statistics and probability as they are applied in a number of problems
3. differentiate the various types of probability distributions and use them to solve probability problems on probability distribution
4. Use sampling theory as a basis in solving hypothetical problems geared towards establishing levels of confidence

MODULE CONTENT
Mathematics of probability. Sample spaces, events, operations with events, counting, sample points, probability of an event, additive rules, conditional probability, multiplicative rules, Baye’s rule (15%)
Distribution of random variables. Concept of a random variable, different probability distributions (discrete, continuous, uniform, binomial and multinomial, hyper geometric, geometric, Poisson, Normal), means and variance of a random variable, properties of mean and variance (40%)
Sampling theory and estimation of parameters. Sampling distribution, t-distributions, sampling procedures, statistical inference, estimating the mean, estimating a proportion, estimating variance, Bayesian methods of estimation, hypothesis testing, regression and correlation (40%)
Nonparametric statistics. Different nonparametric tests (5%)

BCS 216 NUMERICAL ANALYSIS 2 (2L, 1T) UNITS

OBJECTIVS
To derive, apply and appreciate simple numerical techniques for solving some basic problems in numerical analysis

Learning outcomes
At the end of this course the student should be able to
1. Derive and use methods to find roots of functions, perform interpolation and extrapolation, numerical integration and differentiation
2. Use MAPLE or MATLAB programming languages to implement numerical methods that facilitates the numerical solutions of scientific applications and mathematical problems

MODULE CONTENT
SOLUTIONS OF EQUATIONS IN ONE VARIABLE. The bisection method, the fixed-point iteration method, Newton-Raphson method, the secant method (15%)
Interpolation and polynomial approximation. The Taylor polynomials, Lagrange polynomial, interpolation by splines (15%)
Numerical quadrature. Numerical differentiation and integration, composite numerical differentiation and integration, adaptive quadrature methods (20%)
Direct methods for solving linear systems. Linear systems and equations, Gaussian elimination and backward substitution, pivoting strategies (10%)
Initial-value problems for ordinary differential equations. Elementary theory of IVP, Euler’s method, higher-order Taylor methods, Runge-Kutta methods (20%)
Approximation theory. Discrete least squares approximation, orthogonal polynomials and least- squares approximation, Chebyshev polynomials and economisation of power series (20%)

BCS 203: SOFTWARE ENGINEERING 3 (2L, 1P) UNITS
Course objective:
The course aims to extend and deepen the reader's understanding of software development issues for large scale development through modelling and notation. This is achieved through the studies of the Software Life-Cycle and the Unified Modelling Language (UML)

Learning outcomes:
1. Give the reasons for the emergence of the software engineering study area
2. Describe the Software Life-Cycle
3. Use the UML notation and the corresponding diagrams in designing systems

Course contents

**Introduction to UML:** Class diagram, Object Diagram, Use case diagram, Sequence diagram, Collaboration diagram, Interaction diagram, activity diagram, state diagram,

**Requirements:** What are requirements? Classifying the requirements, Characteristics of requirements, Requirements elicitation, Documenting the requirements, Requirements analysis and negotiation, Requirements validation

**Analysis: Object Diagrams & Class Diagrams:** Object Diagrams, Class Diagrams, attributes, methods, relationships, multiplicity, inheritance, abstract classes, association, aggregation, specialization, generalization

**Analysis: Use case and Interactions diagram:** use case, use case diagram, creating use case, scenarios, relationships, use case to sequence diagram, interaction diagram, collaboration diagram

**Analysis: Sequence and activity diagrams:** sequence diagrams, creation/deletion of objects, iterations, selection, activity diagrams, constraints

**Design:** cohesion, coupling, module independence, functional cohesion, sequential cohesion, abstraction, encapsulation, information hiding

**Design:** High level design, user interfaces, storage, low level design

**Finite State Machines:** State diagrams, States, Events, Transitions, Parsing input with state diagrams

**Software Development Process Models:** “Waterfall” model (Linear Sequential Model), Prototyping Development Model, Spiral Model, Rational Unified Process (RUP)

**Ethics:** Reasons for Ethics, Ethics of the IEEE, ACM, How to analyse scenarios, Living with ethics

**LAB WORK.** Students to work on creating small systems using UML diagrams

**READINGS**
1. Schmuller, Teach Yourself UML in 24 Hours, Sams Publishing
2. Ian Sommerville, Software Engineering, Addison Wesley
3. Pooley and Stevens, Using UML for Software Engineering, Addison-Wesley

**BCS 209: ADVANCED SOFTWARE ENGINEERING 2 (2L, 1T) UNITS**
Pre-requisites: BCS 100, BCS 102, BCS 203

**Course objective:**
The course aims to expand the reader's understanding of software development issues for large scale development. The course touches on the quality issues of software.

**Learning outcomes:**
At the end of this course the student should be able to
1. Outline the reasons and need for quality software
2. Distinguish between the types of software quality measures

Course contents


Testing & Debugging. Testing Strategies for Conventional Software and for Object-Oriented Software, Documentation and System Testing, Debugging, debuggers, Testing Techniques, White Box Based, Black Box Based, Object-Oriented Testing

Process and Project Metrics. Metrics: Motivation and Definition, People and Metrics, Quality and Process Improvement, Organizations and Process Improvement

Software Quality. Concept of Quality, Cost of Quality, Software Quality Assurance, Example SQA Activity, Statistical Software Quality Assurance, Software Reliability, Standards

Technical Metrics. Measurement of Quality, Requirements for use of Metrics, Metrics for Different Phases of Development, Analysis of Data


READINGS
1. Schmuller, Teach Yourself UML in 24 Hours, Sams Publishing
2. Ian Sommerville, Software Engineering, Addison Wesley
3. Pooley and Stevens, Using UML for Software Engineering, Addison-Wesley

BCS 103 ALGORITHMS AND DATA STRUCTURES 3 (2L, 1P) UNITS

Objectives
The aim of the course is to introduce the reader to the different data structures and the algorithms that operate on those data structures. Readers will learn how to use pseudocode to create, modify, delete and combine data structures. The application of the data structures will be given on the basis of various algorithms.

Learning outcomes:
At the end of this course the reader should be able to:
1. Define what is a an ADT
2. Describe the linear ADT and their uses and implementations
3. Describe the non-linear ADT including their uses and implementations
4. List down the characteristics of algorithms
5. Explain the complexity types of algorithms
6. Write pseudocode of the various types of sorting, recursive and searching algorithms and test them on a computer

Course content
Linear abstract data types. Queues, dequesues, stacks, linked structures (20 per cent) Trees. General trees, binary trees, tree traversal, binary search trees (15 percent) Graphs. (10 percent)

String processing. Concatenation. Substrings. Matching. (5 per cent.)

Internal searching and sorting. Methods such as binary, radix, shell, quicksort, merge sort. Hash coding. (15 per cent.)

Recursion. (10 per cent.)

Efficiency of algorithms. Worst-case analysis. Average-case analysis. Big O notation, small o notation. P, NP hard and NP complete problems. (15 per cent.)

Lab work. Students should be given homework problems to test some of the algorithms (10 percent)

References

Programming with Data Structures by Robert L. Kruse, Prentice Hall

BCS 104 DISCRETE STRUCTURES 3 (3L, 1T) UNITS

Objectives

The course aims to provide the student with knowledge on Combinatorial enumeration that seeks to describe a counting function which counts the number of objects in \( S_n \) for each \( n \).

Learning outcomes:
At the end of this course the student should be able to:
1. Distinguish between counting combinations and counting permutations
2. Describe the concept of graph theory and state its use in linguistics, chemistry, and operational research
3. Explain the theory/concept of computability (kinds of problems computers can in principle solve)

Module content

Graphs: General graphs, particular classes of graphs; graphs and associated relations; graphs and associated matrices; operations on graphs (homomorphisms, subgraphs....) algorithms on graphs (reachability, closure, spanning tree....); flows in networks; graphs and combinatorial games; operations on trees. (30 per cent).

Boolean algebra. (10 percent)

Finite State Machines. Definition, deterministic, non-deterministic (20 percent)

Computability: [30%]
1. Algorithms and undecidable problems.
2. Unlimited Register Machines (URMs).
3. Closure properties of URM-computable functions.
4. Recursive functions.
5. The Busy Beaver function.
6. Undecidable problems

Combinatorial Enumeration Problems: [20%]
1. Permutations and combinations.
2. The inclusion-exclusion principle.
3. Recurrence relations.

Readings
3. S. Barry Cooper, Computability Theory, Chapman & Hall/CRC, 2003

BCS 201 DATABASE DESIGN 3 (2L, 1P) UNITS

Objectives
To provide students with an awareness of the need for a database system, its role in an organisation and the standard database Management System (DBMS) architecture and to provide them with skills to analyse database requirements and design well-structured relational database designs

Learning outcomes:
At the end of this course the student should be able to
1. Explain what is a database
2. Discuss the advantages and disadvantages of databases and explain the architecture of databases
3. Describe the different approaches to database design and list down the advantages of the relational model design
4. Perform 1NF, 2NF 3NF and BCNF normalisation, ER and EER modelling
5. Design and implement well-structured relational databases

Module content
Review of traditional file processing systems. Batch processing, online processing, updation (grandfather, father, son files). Limitations of traditional file processing, program/data dependence problem (10 percent)

Three level architecture of DBMS. Need for DBMS, functions of DBMS, components of DBMS, program/data independence (10 percent)

Overview of database models. Role of data model, hierarchical data model, network data model, relational model, object-oriented data model (10 percent)

Relational modelling. Basic concepts of table/relation, keys (primary, foreign, candidate), relational constraints (entity, referential, and business constraints) (15 percent)

Mathematics and relational model. Set theory, relational algebra, relational calculus, SQL, QBE (15 percent)

Normalisation. Normal forms (UNF, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF, 6NF) (15 percent)

Conceptual Modelling using Entity relationship Diagrams (ERD). Fact finding techniques, specifying data and user requirements in terms of business rules of an organisation, basic constructs in ERD, translating business rules into ERD constructs (15 percent)

Mapping an ERD into a relational schema. Steps for translating an ERD into a relational schema (10 percent)

LAB WORK. Students to work on implementing databases on ACCES or similar system and drawing ERD using Microsoft Visio or similar package
Textbooks:
2. Date, C J *Introduction to Database Systems*, Addison Wesley

**BCS 204 MIS 2 (2L, 1T) Units**

**Objectives**

To provide students with an understanding of the roles of various types of management information systems in business organisations and to provide them with knowledge of the relationships between the decision-making process, the organisational hierarchy and the supporting management information systems.

**Learning outcomes:**

1. Describe the impact of new technology on business processes.
2. Describe the need and functions of decision support systems (DSS) and distinguish between the different types of decision support systems.
3. Explain how the different types of information systems (IS) for competitive advantage.
4. Explain the need for protecting data in MIS and outline how data in an IS can be protected.

**Module content**

**Impact of new technology on business processes.** E-commerce, e-business and information systems, categories of e-commerce, B2B (10 percent)

**The Decision Making Process.** Managing the organisation, models and roles of management, strategic information systems, the value chain model (10 percent)

**Decision Making and management information systems.** Decision Support Systems (DSS), types of DSS, components of a DSS, DSS applications, Executive Support Systems (ESS), Online Analytical Processing (OLAP) (25 percent)

**Collaborative Decision Making.** Collaborative working, Group Decision Support Systems (GDSS) (10 percent)

**Competitive Advantage as driver for management information systems.** Information systems for competitive advantage, supply chain management (SCM), customer relationship management (CRM), enterprise resource planning (ERP), Business value of information, managing change and implementation (20 percent)

**Security issues.** Systems integrity and risk management, information systems security and control, system vulnerability and abuse, professional issues, codes of conduct (25 percent)

Textbooks:

**BCS 206 PC DIAGNOSTICS & MAINTENANCE 2 UNITS COURSE OBJECTIVES**

To teach the students all the essentials of installing and troubleshooting all types of PC hardware.

**Learning outcomes:**
At the end of the course the students should be able to:
5. Describe the basics of the CPU, the peripherals and local area networking equipments
6. Install and configure PC hardware and system components
7. Support and troubleshoot PC hardware, software and peripherals.

Contents:

Hardware basics
- PC Functions and Components: introduction to the PC; the system case; the Motherboard; CPU/MCP; Clock; Memory; I/O Expansion Bus, Speaker, I/O Ports and Cables, PS2 Ports, Parallel Ports, Serial Ports, USB Ports, SCSI, Network Ports, Audio Ports, IrDA Ports, Joystick/MIDI Ports, Interface Cards, Storage Devices, Hard Drives, Floppy Drives, Optical Disk Storage, Power Supply Unit (PSU), The Display Subsystem, Modem Types.
  - Tools and Equipment
  - Maintenance Toolkit

System configuration
- BIOS: The Function of BIOS, BIOS Services, BIOS Components, BIOS Upgrades, BIOS CMOS Memory and Battery
- BIOS/CMOS Settings
- BIOS/CMOS Setup Programs
- System Resources: What are System Resources? Interrupt Request Line (IRQ), Direct Memory Access (DMA), I/O Addresses (Ports), Memory Addresses, Determining Resources in Use, Resource Conflicts, Setting System Resources, Plug-and-Play, Installing Driver Software

Safety and preventative maintenance
- Cleaning and Preventative Maintenance: Computer Case Maintenance, Cleaning a Mouse, Cleaning a Monitor, Cleaning a Keyboard, Cleaning Drive Heads, Cleaning Power Supplies and Fans.
- High Voltage Equipment: CRT Servicing and Handling, Lasers and High-Power Light Sources, High Voltage Equipment

Installing, upgrading and troubleshooting
- Field Replaceable Units: Disassembling a PC, Input Devices, Storage Devices, Reassembling the PC
- IDE Devices: Installing an IDE Device, Preparing a Hard Disk, File Systems
- Peripheral Devices: Installing a Video Card and Monitor, Installing a Modem, UART Chips

Motherboards and components
- Memory: Memory Types, Memory Packaging, Memory Characteristics.
- Motherboard Architecture: Motherboard Components, Motherboard Form Factors, Bus Architecture, Bus Standards
Printers
- Printer Types: Dot Matrix Printers, Inkjet Printers, Laser Printers
- Printer Connections, Configurations and Troubleshooting: Printer Technologies, Printer Connections, Configuring Printer Drivers, General Troubleshooting

Laptop PCs
- Laptop PCs: Differences Between a Laptop and a Standard PC, Problems with Laptops, Personal Digital Assistants (PDAs).

Textbooks
PC Doctor

BCS 207 STRUCTURE OF PROGRAMMING LANGUAGES 2 (1L, 1P) units

Objectives
To develop an understanding of the organisation of programming languages, and to introduce the formal study of programming language specification and analysis

Learning outcomes:
At the end of the course the student should be able to:
1. Define formal languages and formal grammars and analyse the production rules of a formal grammar
2. Describe the characteristics of context-free languages and understand that programming languages are special constructs of context-free languages and appreciate the use of BNF notation in defining programming language
3. Distinguish between compilation and interpretation of languages
4. Write simple lexical analysers

Module contents
Language definition structure. Concepts of the formal description of syntax and semantics, i.e. basic characteristics of grammars of Chomsky types 3 (finite state) and 2 (context-free) and, in particular, of grammars allowing simple parsing methods like LR(k) and LL(k) grammars, as well as introduction of methods for the specification of the semantics-like attribute grammars. A language like PASCAL as an example. (20 per cent.)

Interpretative languages. Compilation versus interpretation. String processing. Vector processing with language features such as those available in APL. (20 per cent.)

Lexical, syntactic and semantic analysis. Introduction to lexical analysis, including scanning, regular expressions, error recovery, symbol tables. Introduction to parsing including different types of bottom-up and top-down parsing, recursive descent methods, table-driven parsers, error recovery, symbol tables. Introduction to semantics analysis including tree traversal, attribute propagation, operator identification. (10 per cent.)

LAB WORK. Students to work on parser generators like YACC

Textbook:

**BCS 208 COMPUTER GRAPHICS AND MULTIMEDIA 2 (1L, 1P) UNITS**

**OBJECTIVE**
To give students knowledge of a general-purpose graphics system and its use and to understand the specific software and hardware problems in computer graphics.

**Learning outcomes**
At the end of this course the reader should be able to
1. Define what constitutes multimedia
2. Explain how each of the multimedia items are stored and manipulated
3. Describe how 2D and 3D graphics is produced and manipulated.

**CONTENTS**

**Part 1 Introduction to**
- Computer Graphics and Multimedia
- 3D Computer Graphics
- OpenGL
- XHTML
- CSS

**Part 2 Images**
- Introduction, visual perception and color representation
- Coding, compression and digital formats

**Part 3 Video**
- Motion perception, coding, compression, digital formats

**Part 4 Models**
- Illumination models (OpenGL)
- Geometric Transformations

**Part 5 Audio**
- Audio: auditory perception, coding, compression, digital formats

LAB WORK. Students to work on computer graphics packages like OpenGL or developing graphics through 3GL languages like Java

**References**

**BCS 200 NETWORK DESIGN & ADMINISTRATION 1 3 (2L, 1P) UNITS**

**Objectives**
The course objective is to provide conceptual knowledge of Computer Networks Design and Administration. The lectures will focus on the design guidelines for
the main local and global networks components: cables, switches, routers. The networks administration part will cover the topics about configuring the client and server parts for different network services, as well as the topics about network security.

**Learning outcomes:**
At the end of this course the student should be able to:

1. Design a Local Area network using a given topology or combination of topologies
2. Diagnose and remedy problems occurring in a network
3. Configure the Windows-based OS for network applications
4. Create switching and routing tables
5. Perform the diagnostics of the networks, using the specialized Hardware and Software
6. Suggest appropriate measures for improvement of the network.

**CONTENTS**

1. **COMPUTER NETWORK DESIGN** [25%]
   Network design overview
   - Reasoning the need
   - Seeking approval
   Designing a home or small office network
   - Selecting computers
   - Selecting networking protocol
   - Expanding the network
   Designing an internet work
   - Segments and backbones
   - Connecting to remote networks
   - Locating equipment
   Finalizing the design
   2. **WINDOWS NETWORK ADMINISTRATION** [30%]
   Locating Application and data
   - Server-based operating system
   - Server-based applications
   - Storing data files
   Controlling the workstation environment
   - Drive mappings
   - User profiles
   Controlling the workstation registry
   - Using system policies
   - Remote registry editing
   - Windows 2000 group policies

3. **LINUX NETWORK ADMINISTRATION** [45%]
   Domain Name System
   - DNS Address Translations
   - Local Area Network Addressing
   - BIND
   - Domain Name Service Configuration
   - named.conf
   - options Statement
   - Resource Records
Zone Files
- Subdomains and Slaves
- IP Virtual Domains
- Cache File
- Dynamic Update: DHCP and Journal Files
- DNS Security: Access Control Lists, TSIG, and DNSSEC
- Split DNS: Views

**DHCP Server**
- Configuring DHCP Client Hosts
- Configuring the DHCP Server
- Dynamic Addresses
- Dynamic DNS Updates
- Subnetworks
- Fixed Addresses

**NFS and NIS**
- Network File Systems: NFS and /etc/exports
- Network Information Service: NIS

**Samba**
- Samba
- Documentation
- Samba Applications
- Starting Up Samba
- Passwords
- Configuring the Samba with redhat-config-samba
- The Samba smb.conf Configuration File
- SWAT and smb.conf
- Testing the Samba Configuration
  - Domain Logons
- IPv4 and IPv6
  - TCP/IP Network Addresses
  - IPv6 Addressing
  - TCP/IP Configuration Files
  - Domain Name Service (DNS)
- Network Interfaces and Routes: ifconfig and route
- Monitoring Your Network: ping, netstat, tcpdump, and Ethereal

**LAB WORK.** Students to work on configuring and installing networks and practice all aspects of network administration

**Reference:**

1. Craig Zacker, *Networking, The complete* 200
BCS 205 LINUX SYSTEM ADMINISTRATION 2 (1L, 1P) UNITS

Objectives
To enable students acquire knowledge required to run a system with LINUX operating system. That is the functions of a LINUX System administration. It covers all the things that you have to do to keep a computer system in usable order.

Learning outcomes:
At the end of this course the student should be able to:
1. Back up files (and restore them if necessary)
2. Install new programs
3. Create accounts for users (and delete them when no longer needed), making certain that the file system is not corrupted, and so on.

CONTENTS
Part 1: Basic System Administration [15%]
- Superuser Control: the Root User
- System Time and Date
- Scheduling Tasks: cron
- System Runlevels: telinit, initab, and shutdown
- Managing Services
- Red Hat Administration Tools
- System Directories
- Configuration Directories and Files
- System Logs: /var/log and syslogd
- Performance Analysis Tools and Processes
- Grand Unified Bootloader (GRUB)
- Backups

Part 2: Managing Users [15%]
- User Configuration Files
- The Password Files
- Managing User Environments
- Adding and Removing Users with useradd, usermod, and userdel
- Managing Groups
- Controlling Access to Directories and Files: chmod
- Disk Quotas
- Lightweight Directory Access Protocol
- Pluggable Authentication Modules

Part 3: Software Management [14%]
- Software Repositories
- Software Package Types
- Red Hat Package Manager (RPM)
- Installing Software from RPM Source Code Files: SRPMs
- Installing Software from Compressed Archives: .tar.gz
- The Concurrent Versions System: CVS
- Packaging Your Software with RPM
Part 4: File System Management [15%]
- File Systems
- File system Hierarchy Standard
- Journaling
- Mounting File Systems Automatically: /etc/fstab
- Mounting File Systems Directly: mount and umount
- Installing IDE CD-R/RW and DVD R/RW Devices
- Creating File Systems: mkfs, mke2fs, mkswap, parted, and fdisk
- CD-ROM and DVD ROM Recording

Part 5: RAID and LVM [10%]
- Enabling RAID and LVM in the Kernel
- Configuring RAID Devices
- Logical Volume Manager

Part 6: Devices and Modules [15%]
- Device Files
- Device Information: /proc and /etc/sysconfig/hwconf
- Installing and Managing Terminals and Modems
- Input Devices
- PCMCIA Devices
- Installing Sound, Network, and Other Cards
- Sound Devices
- Video, TV, and DVD Devices
- Modules

Part 7: Kernel Administration [16%]
- Kernel Versions
- Kernel Tuning: Kernel Runtime Parameters
- Installing a New Kernel Version
- Precautionary Steps for Modifying a Kernel of the Same Version
- Compiling the Kernel from Source Code
- Important Kernel Configuration Features
- Compiling and Installing the Kernel
- Boot Loader Configurations
- Module RAM
- Disks

LAB WORK. Students to work and practice all aspects of system administration using LINUX

Reference
1. Lars Wirzenius at el, The Linux System Administrator's Guide, Version 0.9

BCS 202 OPERATING SYSTEMS 3 (2L, 1P) UNITS

Course Objectives
The aim of the course is to provide the basic theoretical principles/concepts of operating systems (services offered, memory management, process management, I/O management, security).

Learning outcomes:
Upon completion of this course, the students should be able to:
1. understand the key concepts of modern operating systems;
2. specify the trade-offs of different algorithms in operating system design;
3. apply the knowledge to system performance tuning and system administration;
4. contrast the wide spectrum of operating systems in different platforms;

**COURSE OUTLINE**

1. Background [10%]
   1.1 Overview of computer system and operating system

2. Process Management [20%]
   2.1 Process description and control
   2.2 Process scheduling: Round-robin and priority scheduling
   2.3 Threads, Symmetric Multiprocessing

3. Concurrency and Synchronization [25%]
   3.1 Mutual exclusion and critical section
   3.2 Lost update problem
   3.3 Busy waiting vs. blocking
   3.4 Peterson’s algorithm
   3.5 Interrupt disabling and spinlock
   3.6 Semaphore and its application in various synchronization problems

4. Memory Management [25%]
   4.1 Partitioning, paging and segmentation
   4.2 Virtual memory
   4.3 Address translation and page fault handling
   4.4 Memory management hardware: page table and Translation Lookaside Buffer
   4.5 Memory management algorithms: fetch policy, replacement policy, resident set management and cleaning policy

5. Input / Output Management and Disk Scheduling [10%]
   5.1 I/O devices
   5.2 Organization of I/O function
   5.3 I/O buffering
   5.4 Disk scheduling, RAID

6. File Management [10%]
   6.1 Organization: files and directories
   6.2 Secondary storage management, file systems: FAT and NTFS

**LAB WORK.** Students to work and practice various operating systems (DOS, WINDOWS, LINUX, etc)

**Textbook**

**References**

**BCS 210 PROGRAMMING IN C 3 (2L, 1P)**

**COURSE OBJECTIVE**
The aim of this course is to provide students with knowledge in C programming

**Learning outcomes:**
At the end of the course the student should be able to
1. Describe the C basics such as statements, variables, and functions and the basic syntax of the C language.
2. Write programs that use the control structures and perform input and output using standard devices and files
3. Write programs that require advance features such pointers and structures

**Course contents:**
Introduction. Variables, constants, operators, expressions, Strings manipulating strings, header files, #include, #define, #if comparisons parsing, arrays
Program Control: if and else statements, for, do and while loops, switch statements.
Functions. definitions, prototypes Arrays Declaration, passing to functions sorting Pointers declaration and initialization
File Processing reading, writing, randomly accessing
Structures declaration and initialisation, accessing members, unions, Data Structures linked lists, trees, and queues.

**Text Book:**
The C programming Language by Brian BCS. Kernighan & Dennis M. Ritchie 1995

**BCS 211: Geographical Information Systems GIS 2(1L+1P) Units**
*Prerequisite: IS 161, IS 137, MT 100*

**Course objective:**
To introduce students to the concepts and use of Geographical Information Systems (GIS)

**Course outcomes:**
At the end of the course the student should be able to:
1. Describe applications where GIS is used and to describe the sources of data in GIS applications
2. Explain how current GIS tools and techniques that can be used to solve some specific problems.
Course contents
· Georeferenced Data
  Measuring locations, distances, and areas
  Map scale
  Metadata
  Mapping principles Global Positioning System (GPS)
· Managing Tabular Data
  Data sources
  Data formats
  Querying and selecting data
  Data summaries
  Relational database introduction
  Tables, relations, and normalization.
  Relational data operations: joins and links
  Data handling
  The field calculator: operating on tables as objects
  Controlling errors
  File formats and file conversion
  Data dictionaries
  Examples of tabular data sets
· Global Positioning System (GPS)
· Remote Sensing
· GIS in Location Based Services (LBS)
· Map Projections
· Common Coordinate Systems
· GIS Planning and Implementation
· Development of an Urban and National GIS: Case Studies
· Past, Present, and Future of GIS

Hands-on Sessions Topics (Using ArcGIS or ArcView or similar software):
· Working with Layers and Maps
· Working with Labels and Annotation
· Displaying Locations from Tabular Data
· Modifying ArcGIS Interface
· Designing a GIS Database
· Automating Data
· Editing the Database Schema
· Editing Spatial and Attribute Data
· Data Management
· Spatial Analysis Functions

Textbooks
N. Chrisman, Exploring Geographical Information Systems, Wiley

BCS 213 COMPILERS 2 UNITS

COURSE OBJECTIVE
The aim of this course is to describe the principles of compilation process as a basis for
the implementation of high-level languages.

**Learning outcomes:**
At the end of this text the reader should be able to
1. Define what is a compiler
2. List down the various types of compilers
3. Describe the main phases of the compilation process
4. Explain how and why object code optimisation is performed

**Course contents:**
- **Introduction:** Language translators: compilers and interpreters. The structure of a compiler: lexical analysis, parsing, semantic analysis, intermediate code generation, register allocation, global optimization. Bootstrapping a compiler
- **Lexical scanning:** Token classes, keyword recognition, minimizing the code-per-character cost of scanning, scanning numeric literals and string literals. The interface between the scanner and the parser. Hand-written vs. automatically generated scanners.
- **Formalism:** regular grammars, regular languages, FSA, non-deterministic FSA, automatic generation of lexical scanners.
- **Semantic analysis:** attributes and their computation, tree-traversals, visibility and name resolution. Inherited attributes and symbol tables. Name resolution in block-structured languages.
- **Type checking.** Type systems, varieties of strong typing, overload resolution, polymorphism and dynamic dispatching. Type-checking and type inference, unification.

- **Run-time organization:** storage allocation, non-local references, parameter passing, dynamic storage allocation. Exception handling, debugging information.
- **Intermediate code generation:** control structures, expressions, simple register allocation. Aggregates and other high-level constructs.
- **Code generation over basic blocks.** Using dags. Global register allocation and graph coloring.
- **Global optimization:** data flow analysis, Single-Assignment form.
- **Code generation for RISC machines:** delay slots, instruction scheduling, inlining, loop unrolling.
- **Peephole optimization.**

**References**
- *Compilers: Principles, Techniques, and Tools* is a famous computer science textbook by Alfred V. Aho, Ravi Sethi, and Jeffrey D. Ullman about compiler construction. Although decades have passed since the publication of the first edition, it is widely regarded as the classic definitive compiler technology text
BCS 214 THEORY OF COMPUTATION 2 UNITS

COURSE OBJECTIVE
The theoretical foundations of computer science have expanded substantially in recent years. The objective of this course is to introduce students to this fundamental area of computer science which enables students to focus on the study of abstract models of computation. These abstract models allow the students to assess via formal reasoning what could be achieved through computing when they are using it to solve problems in science and engineering. The course exposes students to the computability theory, as well as to the complexity theory. The course introduces basic computation models and their properties, and the necessary mathematical techniques to prove more advanced attributes of these models.

LEARNING OUTCOMES:
At the end of the course the student should be able to:
1. Answer fundamental questions about problems, such as whether they can or not be computed, and if they can, how efficiently
2. Express computer science problems as mathematical statements and to formulate proofs
3. Be proficient in key topics of theory of computation, and to have the opportunity to explore the current topics in this area.

COURSE OUTLINE:
History and Preliminaries
Finite Automata and Regular Languages: determinism and nondeterminism, checking vs. computing, properties of finite automata, regular expressions, the pumping lemma, closure properties
Universal models of computations: issues of computability, the Turing machine, translation between models, model independence

Computability theory: primitive and partial recursive functions, encoding a turing machine, recursive and R.E. sets, Rice's theorem and the Recursion theorem, unsolvability
Complexity theory: reducibility among problems, reduction and complexity classes, hierarchy theorems, model-independent complexity classes, NP-completeness, space completeness, provably intractable problems
Proving problems hard: NP-complete problems, P-completeness proofs, Turing reductions and search problems, the polynomial hierarchy and enumeration problems
Complexity theory in practice: restriction of hard problems, strong NP-completeness, the complexity of approximation, the power of randomization
Average case complexity, models of parallel computation, communication and complexity, zero knowledge and probabilistically checkable proofs, complexity and constructive mathematics

TEXT BOOK:
The Theory of Computation, Bernard M.Moret. Addison-Wesley
BCS 301 INTERNET PROGRAMMING AND E-APPLICATIONS 2 (1L, 1P) UNITS

Course objectives.
To enable students understand the Internet technologies and to implement E-applications

Learning outcomes:
At the end of this course the student should be able to:
   1. Define most terms associated with Internet services
   2. Design and implement E-applications incorporating tight security features

Course contents
Web Technologies (Browser, HTML, The Dynamic web, Dynamic Server Concepts, Web Application Servers, XML, Plug-ins, VRML)
Security and privacy on the Internet (Cryptography, digital signatures, Client-based Security, Server-based security)
Search engines and portals (Intelligent Network Agents, Portal Sites, All-in-One Mega web sites)
Interactive Communication Experiences (Online meetings, Internet Chat Solutions, Internet- based Training (e-training, e-learning, portals)
LAB WORK. Students should design and implement Internet based applications. Case studies in commerce, government, shopping, etc.

Readings

BCS 302 DATABASE SYSTEMS 3 Units

Course objective
To introduce the student to the important concepts related to the implementation and management of databases.

Learning outcomes:
At the end of this course the student should be able to:
   1. List down the advantages of concurrent transaction processing and state the importance of transactions in database systems
   2. Describe the concepts of locks and locking in relation to concurrent database systems
   3. Show how data integrity is maintained in database systems via recovery and
security measures

Contents:
E-R modeling: First level design, list of transactions, flexing, second level design, mapping into an implementation. (15 percent)

Queries: Formulation, processing, optimization. (10 percent)

Security: Access control – identification and authentication, locking, objects to be locked – data objects, access paths, schema views, programs, communication objects, types of access, crypt-systems, private/public key, digital signatures, special needs of statistical databases, reasons, perturbation techniques. (25 percent)

Concurrency control: transactions – definition, states, database architecture, the three concurrency problems, serialisability – schedules, equivalent schedules, constrained/unconstrained write, locking two phase locking protocol, tree protocol, granularity, deadlocks, time stamp ordering. (20 percent)

Database recovery: types of failures, backups, log-based recovery, shadow paging. (15 percent)

Distributed databases: replication and partition, queries, updates and integrity, failure.(15 percent)

Textbooks:
Elmasri, Ramez & Navathe, Shamkant Fundamentals of Database Systems, Addison Wesley
Date, C J Introduction to Database Systems, Addison Wesley

BCS 303: PROJECT MANAGEMENT – 2 Units

Course objective
The aim of this course is to enable students to understand the principles of dynamic project models, development process control, user, client and other stakeholder interaction and cross-functional teams.

Learning outcomes:
At the end of this course the student should be able to:
1. Define terms associated with projects and project management
2. Give the reasons why project management is important in organisation
3. Compare and contrast the different project management tools
   - (Gantt charts, CPM, PERT, WBS, etc)
4. Use MS Project in managing a project

Course content
Introduction. Project Management (PM) Fundamentals, People, Process, Product, Technology, Classic Mistakes

PM Processes & Organization. Software project phases, Organizational structures, Project charter, Statement of Work (SOW)

Planning Phase. Development lifecycle models, Matching lifecycles to projects, Project plans, Work Breakdown Structures (WBS)

Estimation and Budgeting. Estimation, Budgeting, Project selection, NPV, ROI, Payback models
Scheduling. Project network diagram fundamentals, PERT techniques, Gantt charts, Critical chain scheduling, Mid-term preview. Use of different packages for project scheduling

Risk and Change Management. Risk management, Change control, more on MS Project

Development Management. Team models, Requirements process, Configuration management, Software metrics, Programming languages & tools, Managing conflict and motivating

Project Control. Status reporting, Project metrics, Earned value analysis, Communications Techniques, Process Improvement, MS Project: (a) Resource levelling (b) Other views

System Test Process. Test specifications, Black box and white box testing, Test scripts, Unit and integration testing, Acceptance test specifications, Test tools, MS Project: (a) Reporting

Final Phases & Other Issues. Project Recovery, Documentation, Cutover/Migration, Post Project Reviews, Closing, MS Project: (a) Advanced features

Project Success. Management support, Expectations, Success metrics

References
M J Earl, Management Strategies for Information Technology, Willey 1996

BCS 305 COMPUTER SYSTEM SECURITY 2 units

Course objective
This course aims to teach students the principles of computer security from an applied viewpoint and provides hands-on experience on security threats and countermeasures

Learning outcomes:
At the end of this course the student should be able to:
1.

Course contents
- Code execution vulnerabilities (buffer overflow, sandboxing, mobile code)
- Malware (trojans, viruses, and worms), access control (users, roles, policies), cryptosystems (hashing, signatures, certificates)
- Network security (firewalls, TLS, intrusion detection, VPN)
- Human and social engineering
- Digital rights management

Textbook
Cryptography and Network Security (Principals and Practice) William Stallings; Prentice Hall, NJ, USA 1999
References
Cryptography Theory and Practice by D. Stinson
Basic Methods of Cryptography; Jan C. A. Van-der-Lubbe; Cambridge University Press

BCS 307 OPERATIONS RESEARCH 2 (2L, 1T) UNITS

OBJECTIVES
The course aims to impart to the students the knowledge and application of scientific methods and techniques to decision-making problems. The objective is to select the best alternative, that is, the one leading to the best result.

Learning outcomes:
At the end of this course students should be able to:
1. Classify operations research problems into their categories
2. Solve operations research problems in particular the student should be able to:
   i. formulate the problem
   ii. construct a model of the system
   iii. select a solution technique
   iv. obtain a solution to the problem
   v. establish controls over the system
   vi. implement the solution

Course Contents

<table>
<thead>
<tr>
<th>Course Contents</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction</td>
<td>[7%]</td>
</tr>
<tr>
<td>2. Linear Programming Modelling</td>
<td>[8%]</td>
</tr>
<tr>
<td>3. Graphic Method</td>
<td>[5%]</td>
</tr>
<tr>
<td>4. Simplex Method</td>
<td>[7%]</td>
</tr>
<tr>
<td>5. Duality theory</td>
<td>[6%]</td>
</tr>
<tr>
<td>7. Network Flow Algorithms</td>
<td>[8%]</td>
</tr>
<tr>
<td>8. Integer Programming</td>
<td>[15%]</td>
</tr>
<tr>
<td>9. Dynamic Programming</td>
<td>[15%]</td>
</tr>
<tr>
<td>10. Nonlinear Programming</td>
<td>[20%]</td>
</tr>
</tbody>
</table>

Textbook:

BCS 313 DISTRIBUTED SYSTEMS 2 (2L, 1T) UNITS

Course objectives:
The main objective of this course is to study the design issues involved in distributed computing systems.

Learning outcomes:
At the end of the course the student should be able to
1. Discuss the motivation for building distributed systems and to state the various algorithms and protocols proposed in literature for system operability.
2. Describe some of the experimental distributed systems that have been built in the last few years paying special attention to the fault-tolerant and performance aspects of these systems.

Course content
- Characterization and models [10%]
- IPC/RPC [10%]
- Distributed File Systems and Name Services [12%]
- Time, Global States, and Coordination [10%]
- Replication [10%]
- Shared Data, Transactions, and Conc. Control [10%]
- Java RMI/CORBA/P2P [10%]
- Jini and Enterprise Java Beans [5%]
- SOAP/XML Distributed Multimedia Systems [5%]
- Distributed Operating Systems and Case Studies [5%]
- Security [10%]
- Other Distributed System Technologies [10%]

Readings

BCS 310 Computer Simulation and Modelling 2 (2L, 1P) UNITS

Objectives:
This course will enable students to be competent in the application of computing techniques in solving and describing various real world phenomena and stimulating their creative skills.

Learning outcomes:
1. Demonstrate the understanding of terms and concepts as referred to computer simulation and modelling
2. Use at least one simulation package such as MATLAB, R
3. Formulate and analyse mathematical models from given real world phenomena and design algorithms to solve/simulate the models

Contents
Basic concepts and Terminologies
Definitions, Model Taxonomy, Application areas, advantages and Disadvantages of Simulation, Simulation and Experiment, System and system environment

Simulation Tools (Software Packages)
MATLAB Introduction, Getting started, Basic syntax and Variables, Vectors and matrices operations, Conditions and Loops, Built in functions, Visualisation and M-files, Modelling examples
Model Formulation and Simulation
BCS 311 INTRODUCTION TO ARTIFICIAL INTELLIGENCE 2 (2L, 1P)

UNIT Objectives
This course surveys the issues and techniques involved in the creation of computer systems that engage in intelligent behaviour.

Learning outcomes:
1. Define what is artificial intelligence
2. Chart out the evolution of artificial intelligence from antiquity to the era of digital computers
3. Explain the criterion for determining an intelligent program/machine
4. Show how intelligence databases are built and used using Prolog.

Contents:
1. Introduction to Artificial Intelligence [ 7 %]
2. Intelligent Agents & Problem Solving as Search [ 8 %]
3. Uninformed Search [ 7 %]
4. Informed/Heuristic Search. [ 7 %]
5. Local Search and Constraint Sat Search [ 7 %]
6. Game Playing [ 7 %]
7. Representing Knowledge and Propositional Logic [ 7 %]
8. First Order Logic and Description Logics [ 5 %]
9. Logical Reasoning [ 5 %]
10. Uncertain Reasoning & Basic Probability Theory [ 7 %]
11. Graphical Models [ 8 %]
12. Machine Learning/Induction [ 5 %]
13. Learning with Knowledge [ 6 %]
14. Probabilistic Sequence Processing [ 7 %]
15. Speech and Language Processing [ 7 %]

LAB WORK. Students to work on PROLOG and create AI databases

Readings
The required text for this class is Artificial Intelligence: A Modern Approach by Stuart Russell and Peter Norvig. It should be in the campus bookstore, but you should be able to get a better price online.

BCS 300 SYSTEM ANALYSIS AND DESIGN 3 (2L, 1P) UNITS
Course objective: To impart knowledge to students in system analysis and design to enable the analysis of real life system and design new systems according to user requirements

Learning outcomes
At the end of this course the student should be able to:
1. Define what is a system
2. Explain why systems are broken down into subsystems
3. Describe the methods used in finding the facts of a system
4. Describe the phases in the different methodologies used in designing new systems (SDLC, JAD, RAD, SSADM, and Prototyping)
5. Use the methodologies in developing a new information system.

CONTENTS
Part 1
Introduction to the systems processes 20%
- Systems - Close/open, real/abstract, deterministic/non-deterministic (10 percent)
- Stepwise refinement, top-down design - Modules. Coupling, cohesion (5 percent)
- System development methodologies
  - The Systems Development Life Cycle (SDLC)
  - JAS
  - RAD and PROTOTYPING

Part 2
System Analysis and Modelling 20%
- Preliminary Systems analysis
- Detailed Analysis
- Analysing user requirements
- Specifying new system according to user requirements

Part 3 System Design 35%
- DFDs of new system
- ELH
- ERD
- Database design
- Function specifications
- Design of screen dialogues

LAB WORK. Working to solve real life problems, for example, develop a library system, school management system, supermarket processing system, etc using the above methods (30 percent)

Text book
2. Chris el, Structured System Analysis (tools and Technique), Prentice Hall, INC (1979)

BPT 199 FIRST YEAR PRACTICAL TRAINING 2 UNITS
BPT 299 SECOND YEAR PRACTICAL TRAINING 2 UNITS
Objectives
(a) To expose our students to various research and/or production activities being carried out in different parts of the country.
(b) To enable students to apply or learn the practical applications of the scientific theory they learn at the University in solving real-life problems.
(c) To ensure that our Computer Science graduates leaving the University college will have acquired some appropriate work experience.
(d) To establish and maintain contacts between prospective employers and the FACULTY OF SCIENCE so as to ensure that our students are given the appropriate skills and knowledge for the jobs they are likely to do after graduation.
(e) To enable prospective employers and employees to become acquainted with one another in the work situation. In future it may be possible for this association to continue on a permanent basis after the student graduates.
(f) To help lessen the dangers of intellectual elitism and diminish the wrong idea harboured by some of our students and staff that “book-knowledge” ranks higher than practical work experience.

Module Contents
Each student reading for a degree in ICT will spend time during Practical Training periods each lasting **six to eight** weeks.

Assessment: will be based on **five** items each of which will be marked as follows
(a) Employer supervisor’s assessment 20%
(b) Student’s daily log-book assessment 15%
(c) University supervisor’s report 10%
(d) Student’s final report 40%
(e) Student’s presentation after presenting final report 15%

BCS 302 DATABASE SYSTEMS 3 (2L, 1P) Units

Aim
To introduce the student to the important concepts related to the implementation and management of databases.

Contents:

**E-R modelling:** First level design, list of transactions, flexing, second level design, mapping into an implementation. (15 percent)

**Queries:** Formulation, processing, optimization. (10 percent)

**Security:** Access control – identification and authentication, locking, objects to be locked – data objects, access paths, schema views, programs, communication objects, types of access, crypt- systems, and private/public key, digital signatures, special needs of statistical databases, reasons, and perturbation techniques. (25 percent)

**Concurrency control:** transactions – definition, states, database architecture, the three concurrency problems, serialisability – schedules, equivalent schedules, constrained/unconstrained write, locking two phase locking protocol, tree protocol,
granularity, deadlocks, time stamp ordering. (20 percent) **Database recovery:** types of failures, backups, log-based recovery, shadow paging. (15 percent) **Distributed databases:** replication and partition, queries, updates and integrity, failure. (15 percent)

**Textbooks:**
2. Date, C J *Introduction to Database Systems*, Addison Wesley

**BCS 304 PROFESSIONAL PRACTICES IN INFORMATION SYSTEMS 2**

2 (2L, 1T) UNITS

Course objectives
To make the student understand the technical aspects of information systems and to them the context (ethical, social, legal, and organizational) in which professional Information Systems Practitioners work.

**Learning outcomes:**

**At the end of this course the student should be able to:**

1. Describe components of information system and the technological architectures used in such systems
2. Understand the tasks associated the daily operations (security, service level agreements, daily operations, etc) of information systems
3. Explain the needs for security, use of codes of conduct, intellectual property and data privacy with regard to information systems practice

**Module contents**

**Information systems.** Definition, IT and IS, owners (use, management, exploitation, utilisation), types (centralised, client/server, distributed, two/three tier). Example systems. (15 percent) **Components of IS.** Hardware components. Software (versions, types, copyright, open source). Data sets (protection, privacy and data protection acts, DBMS). Documentation (user/technical manuals. People (programmers, analysts, operators, etc, skills/experience, education levels) (25 percent)

**Management.** Budgeting, supplies, work environment, daily operations, Help desk, system diagnostics/monitoring and tuning. introduction to human resource management. (15 percent)

**Outsourcing.** Reasons of, advantages of, disadvantages of, Service Level Agreements (10 percent) **Security.** Physical security, environment, encryption, backups, disaster recovery. Insurance (20 percent)

**Risk, Security and Control Issues.** Systems integrity and risk management, information systems security and control, system vulnerability and abuse, systems quality problems, disaster recovery and business continuity planning. (15 percent)

**Professional issues.** Data privacy and its management, intellectual property issues, codes of conduct and professional conduct. (10 percent)

**READINGS**

3. M J Earl, Management Strategies for Information Technology, Willey

**BCS 315 Networks Design & Administration II 2 (1L+1P) Units**

**COURSE OBJECTIVE**

To expand the students’ knowledge acquired in BCS 200 and to integrate that knowledge to the CISCO Networking Academy courses to add value to the course.

**Learning outcomes:**

At the end of this course the student should be able to:

1. Analyse for collisions in bridges and switches
2. Configure application layer applications

**Course contents**

Cabling: Wiring closet specifications, Selection practice, Horizontal and backbone cabling, Cabling and grounding, Preparing layer 1 devices (patch panels, RJ45, hubs etc), Cabling troubleshooting.

LAN Extension: Analysis of collision domains bridges and switches, Configuration of switches, troubleshooting of switching devices.

**Routing and addressing design:** IP address classes, creating subnets, Router configuration basics, Adding and configuring routers, Routed and routing protocols.

**Application layer applications:** Configuration of the application layer applications (FTP server, Web server, Mail server etc.), Securing and administering a server, troubleshooting of server software configuration.

**Textbook**


**References**

Gerd E. Keiser, Local Area Networks, McGraw-Hill International editions
Cisco Networking Manual

**BCS 399 FINAL ICT PROJECT 4 UNITS**

**Objectives**

To develop a student’s ability to

1. Plan, organise and work independently on a selected problem drawing on and extending ideas encountered in other study modules
2. Communicate the work carried out, through means of a written report, oral presentation, or and software development.

**Methodology**

The student pursues a non-trivial project or problem in an area of ICT with advice only being provided by the supervisor.

The project can be undertaken by an individual student or by a group of students not exceeding 4 members. At the end of the project a report must be produced.
Duration: 15 weeks starting from second week of first semester in third year.

Assessment: 100% based on the report and presentation. Appendices E and F show how this is done. APPENDIX E is the structure of student presentation evaluation form and APPENDIX F shows the structure of the student report evaluation form. Appendix E shows how student and Appendix F show how this is done.

Text: Weaver, P Success in Your Project: A guide to Student System Development Projects, Pearson Education

BMS 110 INTRODUCTION TO BUSINESS AND MANAGEMENT 2 (2L, 1T) UNITS

COURSE OBJECTIVES
The aim of the course is to provide a thorough understanding of the concepts, principles and practice of management so that the students can understand the application of managerial techniques in decision making for various organizations.

LEARNING OUTCOMES
At the end of this the students should be able to:
1. Use relevant terminologies related to management and organisations
2. Describe the structure of organisations
3. Show how managerial functions such as planning, organising and controlling are interrelated in organisations
4. Show how modern management techniques like Management by objectives and risk management help in making strategic, tactical and operational decisions

COURSE CONTENTS:
1.0 NATURE OF MANAGEMENT, BUSINESS AND ORGANIZATION
   1.1 Nature and Scope of Management
   1.2 Development organization theories
   1.3 Organizations as business units
   1.4 Organizations, Resources and Output

2.0 ORGANISATIONAL THEORY
   2.1 Classical organization theories
   2.2 Modern organization theories

3.0 BUSINESS ENVIRONMENT IN TANZANIA AND ITS IMPACT ON BUSINESS DEVELOPMENT
   3.1 Political environment
   3.2 Economic environment
   3.3 Social/Cultural environment
   3.4 Technological environment

4.0 FUNCTIONS OF MANAGEMENT:
4.1.1 PLANNING
- Definition and Importance of Planning.
- The planning process and types of plans.
- Barriers to effective planning
- Fundamentals of decision-making and barriers to decision making.
- The relationship between decision making and planning.
- Individual vs. group decision making

4.1.2 ORGANIZING
- Meaning and importance of organizing
- Types of organizational structures
- Formal and informal organizations
- Principles effective organizing.
- Span of management, power, authority, responsibility and delegation
- Departmentation
- Centralization and decentralization
- Delegation, Responsibility and Accountability.

4.1.3 DIRECTING/LEADING
- Meaning and importance of leadership
- Leadership theories
- Trait Approach to leadership.
- Behavioural Approach to leadership
- System and Contingency theories of leadership
- Flexibility in leadership

4.1.4 MOTIVATION
- Meaning and importance of motivation in an organization
- Motivation theories: classes
- Reinforcement theory
- Need theory
- Expectancy theory

4.1.5 COMMUNICATION
- Fundamentals and purpose of communication
  1. Communication effectiveness in organizations
     a. Manager communication skills
     b. Barriers to effective Communication
  2. Improving Communication in

4.1.6 Coordinating
- Definition and Importance
  a. Coordination and Cooperation
  b. Coordination and managerial functions
- Pre requisites for achieving coordination
- Problems in Coordination
  a. Techniques used to achieve Coordination

4.1.7 CONTROLLING
- Definition and importance of controlling
- Basic components of the control process
- Characteristics of effective control
- Types of controls
- Problems of Controls
6. an organization

READINGS

BMS 222 SMALL BUSINESS AND ENTREPRENEURSHIP 3 (3L, 1T) UNITS

COURSE OBJECTIVE:
The course introduces to the students the contextual aspects of small businesses and entrepreneurship. Management of Small Business is seemingly demanding in the walks of the contemporary fast changing business environment, since it caters for the needs of job creations to cushion out the alarming increase of unemployment rate as well as arresting fast changing market tastes. Topics covered include; definition, problems, factors influencing entrepreneurship, role of small firms in Tanzania, public polices on small businesses in Tanzania, gender and entrepreneurship etc

Learning outcomes:
At the end of the course the student should be able to:
1. Explain the concepts of small businesses and entrepreneurship and their role in the social economy.
2. Understand the factors influencing entrepreneurial behaviour and small firms start ups and business performance.
3. Participate effectively in building economic capacity through increased productivity through creation of jobs using with tools, skills and techniques learned in the course

COURSE CONTENTS:
1.0 Introduction
   1.1 Small business and the entrepreneur
   1.2 Entry modes into small business
   1.3 Small business regulations
   1.4 Prospects and barriers if small business development in Tanzania.
2.0 Business Concept and business establishment
2.1 Market analysis and Forecasting
2.2 Market search techniques and environmental analysis
2.3 Demand Analysis
2.4 Competitor/Supplier Analysis
2.5 Sales Forecast
2.6 Market Share estimation
3.0 Market Analysis
3.1 Development of a market plan
3.2 Developing marketing strategies
4.0 Business Planning
4.1 Production Plan
4.2 Operations plan
4.3 Financial Plans
5.0 Projected Financial Statements
5.1 Income Statement
5.2 Balance Sheet
5.3 Cash Flow
6.0 Benefits and Risks of Business
7.0 Limitations to Small Business and Entrepreneurship in Tanzania
8.0 The role of Government in Promoting Small Business and Entrepreneurship.

READINGS:
   Occasional Paper Series No. 4 Department of Marketing, FCM, UDSM.
   Suggestion for the future” Research Report submitted to the USAID
4.0 ACADEMIC REGULATIONS

4.1 TEACHING AND LEARNING FACILITIES AND RESOURCES

To run any curriculum you need resources. The resources required differ from case to case. In our case the following are required.

4.1.1 EQUIPMENT REQUIREMENTS
Clearly, some type of computing equipment should be available for the students and faculty to use. The college has ONE computer labs with 20 computers for use by students and has connections to the Internet. Thirty (30) more computer will be installed in July 2015. The University College has internet connection via a local area network and wireless.

4.1.2 LIBRARY SERVICES
The modules include reference/textbooks which are available. However, the Marian University College Library is equipped with both hardcover book and e-book.

4.3 ACADEMIC STAFF FOR THE FIRST ACADEMIC YEAR 2015/2016
The following is a list of academic and technical staff recruited by MARUCO to run the degree programme.

<table>
<thead>
<tr>
<th>S/No.</th>
<th>Name</th>
<th>Academic Qualification</th>
<th>Academic Rank</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Computer Science</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Mr. Antony Mwombeki</td>
<td>BSc Comp Sci</td>
<td>Assistant Lecturer</td>
<td>Full-Time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSc Comp Sci</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Mr. Aldof R. Kamuzora</td>
<td>BSc Comp Sci</td>
<td>Assistant Lecturer</td>
<td>Full-Time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSc Comp Sci</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Mr. William V. Meja</td>
<td>BSc Info System Mangt</td>
<td>Tutorial Assistant</td>
<td>Full-Time</td>
</tr>
<tr>
<td></td>
<td><strong>Mathematics</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Ms. Winfrida Mwigilwa</td>
<td>BSc Education</td>
<td>Assistant Lecturer</td>
<td>Full-Time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSc Math Modelling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Mr. Josephat Itambu</td>
<td>BSc Maths</td>
<td>Assistant Lecturer</td>
<td>Full-Time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSc Maths and Application</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>IT Technician</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Mr. Kilango Dole</td>
<td>Certificate in Computer Sci and Info System</td>
<td>IT Technician</td>
<td>Full-Time</td>
</tr>
<tr>
<td>2</td>
<td>Ms. Lilian Sylvester</td>
<td>BSc Computer Science</td>
<td>IT Technician</td>
<td>Full-Time</td>
</tr>
</tbody>
</table>

NB: University wide courses (Social and Professional Ethics, Communication skills, Development Studies) are taught by staff from other faculties or part-time staff.


4.4 ADIMMISION REQUIREMENTS

Entry Requirements for the Bachelor of Science in Computer Science (Information Systems) Degree

a) Candidates must be holders of Advanced Certificate of Secondary Education Examination (A.C.S.E.E.) with at least two (2) principal passes in relevant subjects AND they must have at least five (5) credit passes in the Certificate of Secondary Education Examination (C.S.E.E. Ordinary Level), including Mathematics and English. In addition, the applicant must have secured in the Advanced Certificate of Secondary Education Examination (A-Level) a total of 4.5 points or more in Mathematics, Physics and Computer Studies (both from the same sitting).

The points are based on the following scale A=5, B=4, C=3, D=2, E=1, S=0.5, F=0

b) OR Candidates must hold qualifications equivalent to the above requirements from institutions recognised by the University

c) OR Candidates must hold a relevant Diploma of at least second class standing from institutions recognised by the University.

4.5 COURSE COMPLETION AND GRADING SYSTEM

1. The faculty of Science operates a semester system of studies. Each semester is 15 weeks long plus two weeks of examinations.

2. The duration of all courses is three years consisting of six semesters plus two Practical Training periods each 6 to 8 weeks long.

3. Each course is given a unit weighting according to the time devoted to it on the timetable.

4. Each course will be examined upon completion at the end of the semester. The pass mark in each examinable course is C. The course work assessment includes papers, assignments, tests, quizzes etc and normally accounts for 40% of the final marks in each course. The end of semester examination accounts for 60% of the total final marks.

5. A minimum of 131 units must be passed for the award of a degree. Each student must include and pass in her/his study programme at least 20 units in the 300 course series in the major subject and not more than 30 in the 100 course series

6. Candidates are required to pass all the subjects of the examinations at the end of each year before proceeding to the next year of study.

7. Pass mark in each of the courses taken shall be C. The mark for “Bad fail” shall be E.

8. A failing candidate may be allowed to take a supplementary examination in the failed subjects provided the overall average in all courses taken in that year constitutes a GPA greater than or equal to 2.0. Otherwise the candidate shall be discontinued, (if in first year); if in second or third year, with an aggregate GPA of 2.0 or above, the candidate shall repeat the year, except that no candidate shall repeat any year more than once.

9. A pass in Supplementary Examination shall be recorded as pass mark, that is, C.

0. The grade to marks correspondence shall be as follows

<table>
<thead>
<tr>
<th>Marks (%)</th>
<th>Grade</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>70-100</td>
<td>A</td>
<td>Excellent</td>
</tr>
<tr>
<td>60-69</td>
<td>B+</td>
<td>Very Good</td>
</tr>
<tr>
<td>Grade</td>
<td>Mark Range</td>
<td>Grade</td>
</tr>
<tr>
<td>-------</td>
<td>------------</td>
<td>-------</td>
</tr>
<tr>
<td>50-59</td>
<td>B</td>
<td>Good</td>
</tr>
<tr>
<td>40-49</td>
<td>C</td>
<td>Pass</td>
</tr>
<tr>
<td>35-39</td>
<td>D</td>
<td>Fail</td>
</tr>
<tr>
<td>0-34</td>
<td>E</td>
<td>Bad fail</td>
</tr>
</tbody>
</table>

- A candidate’s final standing in his/her diploma will be determined by the weighted average for all First, Second and Third year marks.
- The final degree shall be classified as First Class, Upper Second Class, Lower Second Class or Pass. These categories shall be awarded as follows:
  - First class: 4.4 – 5.0
  - Upper second: 3.5 – 4.3
  - Lower second: 2.7 – 3.4
  - Pass: 2.0 – 2.6

### 4.6 ADMISSIONS AND EXAMINATION REGULATIONS
Admissions, examination regulations, academic procedures including disciplinary action and mode of appeal will be as stated in the university college prospectus.

### 4.7 REGULATIONS FOR PROJECT/FIELD WORK
Every student pursuing a degree course, diploma or certificate in ICT will be required to undertake practical training/attachment before he/she graduates. The duration of the course is 6 to 8 weeks depending on what the university decides.

For certificate student the training will take place after the two teaching semesters. For diploma students the training takes place after the two semesters of the first year and for degree students they will do it two times. That is, at the end of the first and second semesters.

The success of this Practical Training Programme will very much depend on the degree of co-operation and support the FACULTY OF SCIENCE will receive from the students and staff in the FACULTY OF SCIENCE as well as from the institutions that at which the students will be placed.

It is expected that our national institutions will assist us in our efforts of training the nation’s high-level manpower in this way.

#### 4.7.1 AIMS AND OBJECTIVES OF FIELD ATTACHMENT
(a) To expose our students to various research and/or production activities being carried out in different parts of the country.
(b) To enable students apply or learn the practical applications of the scientific theory they learn at the University in solving real-life problems.
(c) To ensure that our Computer Science graduates leaving the University college will have acquired some appropriate work experience.
(d) To establish and maintain contacts between prospective employers and the FACULTY OF SCIENCE so as to ensure that out students are given the appropriate skills and knowledge for the jobs they are likely to do after graduation.
(e) To enable prospective employers and employees to become acquainted with one another in the work situation. In future it may be possible for this association to continue on a permanent some of our students and staff that “book-knowledge” ranks higher than practical work experience basis after the student graduates.
(f) To help lessen the dangers of intellectual elitism and diminish the wrong idea harboured by

4.7.2 NATURE OF THE TRAINING

It is the hope of the Faculty of Science that the practical training will:
(i) Be related to the student’s University studies, enabling him/her to see the relation between the theory of his/her studies and the practice of work for development.
(ii) Be related to the student’s future career, enabling him/her to gain on-the-job experience. It is realized, however, that these two objectives may not be attainable for all students, particularly for those students who will have only completed their first academic year at the University. However, the second of these aims can be interpreted in its broader sense, bearing in mind that it is always advantageous for prospective high-level manpower to have work experience at all levels not excluding manual work.

4.7.3 FINANCIAL ASPECTS OF THE PROGRAMME

The practical training period is part and parcel of the student’s University studies. It is an essential component of the Degree Programme, and must be completed satisfactorily before a computer science degree is awarded. For this reason the programme will be fully financed by the student Sponsor. All students who will be involved in the programme will receive their normal financial allowances from the Sponsor. The employer will not be required to pay the student any salary or wages.

4.7.4 METHOD OF ASSESSMENT OF FIELD PRACTICAL TRAINING

Assessment will be on grading similar to other courses.
1) A student must get a pass on both practical training periods before qualifying for a degree.
2) Assessment will be based on five items each of which will be graded as follows a. Employer supervisor’s report 20%
   b) Student’s daily log-book 15%
   c) University supervisor’s report 25%
   d) Student’s final report 40%
3) A student who fails will have to pass a supplementary training period during the following year.
5.0 APPENDICES

APPENDIX A

MARIAN UNIVERSITY COLLEGE
FACULTY OF NATURAL AND APPLIED SCIENCES
BACHELOR OF SCIENCE IN COMPUTER SCIENCE

BPT 199 & BPT 299 PRACTICAL TRAINING
EMPLOYER ASSESSMENT FORM

Name of the student: ________________________________
Year of study (1st or 2nd year) __________________________
Name of Institution: ________________________________
Address: __________________________________________
Date of student’s employment: _________________________
Broad description of the type of work done by the student:
________________________________________________________________________________________
________________________________________________________________________________________

Confidential Assessment of the student by the Employer

Please rank the student by ticking the corresponding column. Use the following rank choices
5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor and comment if necessary.

<table>
<thead>
<tr>
<th>Punctuality in reporting to work</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Consistency in putting in a full day's work</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Initiative and curiosity to learn from work experience</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ability to learn new ideas</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Willingness to engage in any task assigned</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Readiness to co-operate with other members of staff</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ability to take on a task or problem in some depth</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Diligence in work</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Leadership qualities</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Would you want to employ the same student during next year’s practical training period?
Yes/No

Comment:
________________________________________________________________________________________
________________________________________________________________________________________

Signature: ..............................................

Position: ..................................................
APPENDIX B

MARian UNIVERSITY COLLeGE
FACULTY OF NATURAL AND APPLIED SCIENCES
BACHELOR OF SCIENCE IN COMPUTER SCIENCE

BPT 199 & BPT 299 PRACTICAL TRAINING

COLLEGE SUPERVISOR ASSESSMENT FORM

NAME OF SUPERVISOR _____________________________

DATE AND TIME OF VISIT ______________________________

NAME OF INSTITUTION ________________________________

P.O. BOX ___________________ TEL NO. ________________

CONTACT PERSON(S) MET: ____________________________

POSITION ____________________

TRAINING OFFICER(S) ________________________________

POSITION ________________

GENERAL DESCRIPTION OF THE ACTIVITIES OF THE INSTITUTION:

_________________________________________________________________________________

STUDENT(S) NAME AND YEAR: __________________________

DESCRIPTION OF ACTIVITIES BY THE STUDENT: __________________________

Please rank the student by ticking the corresponding column. Use the following rank choices
5 = Excellent, 4 = Very Good, 3 = Good, 2 = Satisfactory, 1 = Poor, 0 = Very Poor
and comment if necessary.

<table>
<thead>
<tr>
<th>Views of local supervisors on the Student</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student’s views of the place as training agency</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Your assessment of the training quality</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suitability of the place for future training</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SPECIAL PROBLEMS OF STUDENT AND/OR INSTITUTION

IS THE PLACE SUITABLE FOR FUTURE PT?: YES/NO __________

REMARKS: ____________________________________________

Name & Signature: _________________________________
APPENDIX C

MARIAN UNIVERSITY COLLEGE

FACULTY OF NATURAL AND APPLIED SCIENCES

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

BPT 199, RPT 299, BBA 399 & BCS 399 STUDENT PRESENTATION

STUDENT PRESENTATION ASSESSEMENT FORM

The interview to be conducted by at least three staff, each scoring the student independently in each of the categories shown below. The average in each category is mark given to the student. The penal to be chaired by either the Supervisor or PT coordinator depending on the nature of the report.

Name of student: __________________________________________
Reg. No.__________________________
Year: ___________________________
Date of Interview: _______________
Name of Report: ____________________________________________

Interview Assessment

<table>
<thead>
<tr>
<th>Item</th>
<th>Mark %</th>
<th>Max score %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appearance</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Confidence</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Command of English</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Use of presentation</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Knowledge of subject</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Question/Answer</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Total score</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

Name & Signature of Chairman: __________________________________________

51
APPENDIX D

MARIAN UNIVERSITY COLLEGE

FACULTY OF NATURAL AND APPLIED SCIENCES
BARCHOLAR OF SCIENCE IN COMPUTER

BPT 199, BPT 299, BBA 399 & BCS 399 STUDENT REPORT STUDENT REPORT ASSESSEMENT FORM

The student’s final PT report to be marked by the PT Coordinator and the Supervisor independently and scored in each of the categories shown below. The average in each category is mark given to the student.

Name of student: ___________________________ Reg.
No._________________ 
Year:_____________________
Date of Interview: ________________
Place of Placement: _______________________________

<table>
<thead>
<tr>
<th>Student Assessment Report</th>
<th>Mark %</th>
<th>Max score %</th>
</tr>
</thead>
<tbody>
<tr>
<td>General structure of the report</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Contents page</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Introduction/problem identification</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Main body (methods used, results analysis)</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>References</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Appendices</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Total score</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

Name & Signature of PT Co-ordinator:

_________________________________________
APPENDIX E

MARIAN UNIVERSITY COLLEGE

FACULTY OF NATURAL AND APPLIED SCIENCES

BARCHOLAR OF SCIENCE IN COMPUTER

BPT 199 & BPT 299 STUDENT LOG-BOOK

STRUCTURE OF STUDENT LOG-BOOK – WEEK SUMMARY

Week start date: 
Week end date: 
Summary of week of day’s activities:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>

New knowledge gained:

New skills gained:

Student’s signature: 
Supervisor’s signature:
APPENDIX F

MARIAN UNIVERSITY COLLEGE

FACULTY OF SCIENCE
BARCHOLAR OF SCIENCE IN COMPUTER SCIENCE

BPT 199 & BPT 299 STUDENT LOG-BOOK

STUDENT LOG-BOOK ASSESSMENT FORM
The student’s log book and week summaries to be marked by the PT Coordinator and the Supervisor independently and scored in each of the categories shown below. The average in each category is the mark given to the student.
Name of student: ____________________________
Reg. No. ____________________________
Year:
__________________________

Date of Interview:

__________________________
Place of Placement: ____________________________

Log book Assessment

<table>
<thead>
<tr>
<th>Item</th>
<th>Mark %</th>
<th>Max score %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear description of activities</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>Completeness of the log book</td>
<td></td>
<td>40</td>
</tr>
<tr>
<td>Skills obtained by the student</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>Knowledge obtained by the student</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

Name & Signature of PT Co-ordinator: __________________________________________________________________________

________________________________________________________________________

91