Vidyasagar University

Curriculum for B.Sc. Honours in Computer Science

[Choice Based Credit System]

Semester-I

Sl.No.	Name of the Subject	Nature	Code	Teaching Scheme in hour per week			Credit	Marks
				L	T	P		
C1	**C1T:** Programming Fundamentals using C/C++	Core Course-1		4	0	0	6	75
	C1P: Programming Fundamentals using C/C++ Lab(Practical)	Core Course1 [Practical]		0	0	4		
C2	**C2T:** Computer System Architecture	Core Course-2		4	0	0	6	75
	C2P: Computer System Architecture(Practical)	Core Course-2 [Practical]		0	0	4		
GE-1	GE-1	GE					4/5	75
	GE-1	GE					2/1	
AECC	English	AECC					2	50
				Total Credits =20				

L=Lecture, T=Tutorial, P=Practical

AECC- Ability Enhancement Compulsory Course: English /Modern Indian Language

Interdisciplinary/Generic Elective (GE) from other Department

[**Four papers are to be taken and each paper will be of 6 credits**]:

[Papers are to be taken from any of the following discipline **(GE-1 Mathematics)]: Mathematics/Physics/Chemistry/Statistics/Geology/Electronics/Economics/Zoology/Botany/Physiology**

Semester-I
Core Courses

Core-I

CC-1 : Programming Fundamentals using C/C++ Credits 06

C1T1 : Programming Fundamentals using C/C++ Credits 04

Theory: 60 Lectures

1. Introduction to C and C++ (3 Lectures)

History of C and C++, Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C++.

2. Data Types, Variables, Constants, Operators and Basic I/O (5 Lectures)

Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf(), cin, cout), Using Basic Header Files (stdio.h, iostream.h, conio.hetc).

3. Expressions, Conditional Statements and Iterative Statements (5 Lectures)

Simple Expressions in C++ (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

4. Functions and Arrays (10 Lectures)

Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.

Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays

5. Derived Data Types (Structures and Unions) (3 Lectures)

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

6. Pointers and References in C++ (7 Lectures)

Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with

Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring and initializing references, Using references as function arguments and function return values

7. Memory Allocation in C++ (3 Lectures)

Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, use of new and delete operators, storage of variables in static and dynamic memory allocation

8. File I/O, Preprocessor Directives (4 Lectures)

Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros

9. Using Classes in C++ (7 Lectures)

Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables &Functions, Objects as parameters, Specifying the Protected and Private Access, Copy Constructors,Overview of Template classes and their use.

10. Overview of Function Overloading and Operator Overloading (5 Lectures)

Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators)

11. Inheritance, Polymorphism and Exception Handling (8 Lectures)

Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Basics Exceptional Handling (using catch and throw, multiple catch statements), Catching all exceptions, Restricting exceptions, Rethrowing exceptions.

Reference Books

1. HerbtzSchildt, "C++: The Complete Reference", Fourth Edition, McGraw Hill.2003
2. BjarneStroustrup, "The C++ Programming Language", 4th Edition, Addison-Wesley , 2013.
3. BjarneStroustroup, "Programming -- Principles and Practice using C++", 2nd Edition, Addison-Wesley 2014.
4. E Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill Education, 2008.
5. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice Hall, 2011.

   5. John R. Hubbard, "Programming with C++", Schaum's Series, 2nd Edition, 2000.
6. Andrew Koeni, Barbara, E. Moo, "Accelerated C++", Published by Addison-Wesley , 2000. 7. Scott Meyers, "Effective C++", 3rd Edition, Published by Addison-Wesley, 2005.
8. Harry, H. Chaudhary, "Head First C++ Programming: The Definitive Beginner's Guide", First Create space Inc, O-D Publishing, LLC USA.2014
9. Walter Savitch, "Problem Solving with C++", Pearson Education, 2007.
10. Stanley B. Lippman, JoseeLajoie, Barbara E. Moo, "C++ Primer", Published by Addison-Wesley, 5th Edition, 2012

C1 P1 : Programming Fundamentals using C/C++ Lab Credits 02

Practical: 60 Lectures

1. WAP to print the sum and product of digits of an integer.

2. WAP to reverse a number.

3. WAP to compute the sum of the first n terms of the following series

   S = 1+1/2+1/3+1/4+……

4. WAP to compute the sum of the first n terms of the following series

   S =1-2+3-4+5…………….

5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.

6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

7. WAP to compute the factors of a given number.

8. Write a macro that swaps two numbers. WAP to use it.

9. WAP to print a triangle of stars as follows (take number of lines from user):

```
    *
   ***
  *****
 *******
*********
```

10. WAP to perform following actions on an array entered by the user:
    i) Print the even-valued elements
    ii) Print the odd-valued elements
    iii) Calculate and print the sum and average of the elements of array
    iv) Print the maximum and minimum element of array
    v) Remove the duplicates from the array
    vi) Print the array in reverse order

    The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

11. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

12. Write a program that swaps two numbers using pointers.

13. Write a program in which a function is passed address of two variables and then alter its contents.

14. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.

15. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc () / calloc () functions or new operator.

16. Write a menu driven program to perform following operations on strings:

a) Show address of each character in string
b) Concatenate two strings without using strcat function.
c) Concatenate two strings using strcat function.
d) Compare two strings
e) Calculate length of the string (use pointers)
f) Convert all lowercase characters to uppercase
g) Convert all uppercase characters to lowercase
h) Calculate number of vowels
i) Reverse the string

17. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

18. WAP to display Fibonacci series (i) using recursion, (ii) using iteration

19. WAP to calculate Factorial of a number (i) using recursion, (ii) using iteration

20. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.

21. Create Matrix class using templates. Write a menu-driven program to perform following Matrix
    operations (2-D array implementation):
    a) Sum b) Difference c) Product d) Transpose

22. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).

23. Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.

24. Create a class Box containing length, breath and height. Include following methods in it:
    a) Calculate surface Area
    b) Calculate Volume
    c) Increment, Overload ++ operator (both prefix & postfix)
    d) Decrement, Overload -- operator (both prefix & postfix)
    e) Overload operator == (to check equality of two boxes), as a friend function
    f) Overload Assignment operator
    g) Check if it is a Cube or cuboid

Write a program which takes input from the user for length, breath and height to test the above class.

25. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.

26. Write a program to retrieve the student information from file created in previous question and print it in following format:

    Roll No. Name Marks

27. Copy the contents of one text file to another file, after removing all whitespaces.

28. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.

29. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

Core-II

CC-2 : Computer System Architecture Credits 06

C2T2 : Computer System Architecture Credits 04

Theory: 60 Lectures

1. Introduction (20 lectures)

Logic gates, boolean algebra, combinational circuits, circuit simplification, flip-flops and sequential circuits, decoders, multiplexers, registers, counters and memory units.

2. Data Representation and Basic Computer Arithmetic (10 lectures)

Number systems, complements, fixed and floating point representation, character representation, addition, subtraction, magnitude comparison, multiplication and division algorithms for integers

3. Basic Computer Organization and Design (8 lectures)

Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, Organization of a basic single-bus computer system.

4. Central Processing Unit (10 lectures)

Register organization, arithmetic and logical operations, Instruction formats, addressing modes, instruction codes, machine language, assembly language, RISC, CISC architectures, Hardwired and micro programmed control unit design.

5. Memory Organization (6 lectures)

Memory interfacing and addressing, cache memory organization.

6. Input-Output Organization (6 lectures)

Input / Output: External Devices, I/O Modules, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels.

Recommended Books:

1. M. Mano, Computer System Architecture, Pearson Education 1992
2. W. Stallings, Computer Organization and Architecture Designing for Performance, 8 Edition, Prentice Hall of India,2009
4. M.M. Mano , Digital Design, Pearson Education Asia,2013
5. Carl Hamacher, Computer Organization, Fifth edition, McGrawHill, 2012.

C2P2 : Computer System Architecture **Credits 02**

Lab Practical: 60 Lectures

A. Digital Experiment

1. Design and implement a full adder circuit using NAND gates only.
2. Design and implement a J. K. flip-flop.
3. Design and implement a 4 bit adder using flip-flop.
4. Design and implement a 4 bit synchronous counter.
5. Design and implement a 8:1 multiplexer.
6. Design and implement a D flip-flop.
7. Design and implement a half subtractor using NAND gates only.
8. Design and implement a 3×8 decoder.
9. Design and implement a 8 bit parity generator.
10. Design and implement a two bit digital comparator.

B. Create a machine based on the following architecture:

Register Set

IR	DR	AC	AR	PC	FGI	FGO	S	I	E
0 15	0 15	0 15	011	011	1 Bit	1 Bit	1 Bit	1 bit	1 Bit

Memory 4096 words 16 bits per word	Instruction format

0 3	4 15
Opcode	Address

Basic Computer Instructions

Memory Reference			Register Reference		Input-Output		
Symbol	Hex		Symbol	Hex	Symbol	Hex	
AND	0xxx	Direct Addressing	CLA	E800	INP	F800	
ADD	2xxx		CLE	E400	OUT	F400	
LDA	4xxx		CMA	E200	SKI	F200	
STA	6xxx		CME	E100	SKO	F100	
BUN	8xxx		CIR	E080	ION	F080	**Optional**
BSA	Axxx		CIL	E040	IOF	F040	
ISZ	Cxxx		INC	E020			
AND_I	1xxx	Indirect Addressing	SPA	E010			
ADD_I	3xxx		SNA	E008			
LDA_I	5xxx		SZA	E004			
STA_I	7xxx		SZE	E002			
BUN_I	9xxx		HLT	E001			
BSA_I	Bxxx						

ISZ_I	Dxxx		

Refer to Chapter-5 of Morris Mano for description of instructions.

3. Create the micro operations and associate with instructions as given in the chapter (except interrupts). Design the register set, memory and the instruction set. Use this machine for the assignments of this section.

4. Create a Fetch routine of the instruction cycle.

5. Simulate the machine to determine the contents of AC, E, PC, AR and IR registers in hexadecimal after the execution of each of following register reference instructions:

a. CLA
b. CLE
c. CMA
d. CME
e. CIR
f. CIL
g. INC
h. SPA
i. SNA
j. SZA
k. SZE
l. HLT

Initialize the contents of AC to $(A937)_{16}$, that of PC to $(022)_{16}$ and E to 1.

6. Simulate the machine for the following memory-reference instructions with I= 0 and address part = 082. The instruction to be stored at address 022 in RAM. Initialize the memory word at address 082 with the operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the execution.

a. ADD
b. AND
c. LDA
d. STA
e. BUN
f. BSA
g. ISZ

7. Simulate the machine for the memory-reference instructions referred in above question with I= 1 and address part = 082. The instruction to be stored at address 026 in RAM. Initialize the memory word at address 082 with the value 298. Initialize the memory word at address 298 with operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the execution.

8. Modify the machine created in Practical 1 according to the following instruction format:

Instruction format

0 2	3	4 15
Opcode	I	Address

a. The instruction format contains a 3-bit opcode, a 1-bit addressing mode and a 12-bit address. There are only two addressing modes, I = 0 (direct addressing) and I = 1 (indirect addressing).

b. Create a new register I of 1 bit.

c. Create two new microinstructions as follows :

   i. Check the opcode of instruction to determine type of instruction (Memory Reference/Register Reference/Input-Output) and then jump accordingly.
   ii. Check the I bit to determine the addressing mode and then jump accordingly.

Generic Elective Paper

GE-I (Interdisciplinary for other department)

GE-I: Computer Fundamentals **Credits 06**

GE-I T1 : Computer Fundamentals **Credits 04**

Theory: 60 lectures

Introduction: Introduction to computer system, uses, types. **6L**

Data Representation: Number systems and character representation, binary arithmetic **12L**

Human Computer Interface: Types of software, Operating system as user interface, utility programs **6L**

Devices: Input and output devices (with connections and practical demo), keyboard, mouse, joystick, scanner, OCR, OMR, bar code reader, web camera, monitor, printer, plotter **10L**

Memory: Primary, secondary, auxiliary memory, RAM, ROM, cache memory, hard disks, optical disks **6L**

Computer Organisation and Architecture: C.P.U., registers, system bus, main memory unit, cache memory, Inside a computer, SMPS, Motherboard, Ports and Interfaces, expansion cards, ribbon cables, memory chips, processors. **12L**

Overview of Emerging Technologies: Bluetooth, cloud computing, big data, data mining, mobile computing and embedded systems. **8L**

Reference Books:

1. A. Goel, Computer Fundamentals, Pearson Education, 2010.
2. P. Aksoy, L. DeNardis, Introduction to Information Technology, Cengage Learning, 2006
3. P. K.Sinha, P. Sinha, Fundamentals of Computers, BPB Publishers, 2007

GE-I P1: Computer Fundamentals Lab **Credits 02**
Practical: 60 lectures

Practical exercises based on MS Office/ Open Office tools using document preparation and spreadsheet handling packages.

MS Word

1. Prepare **a grocery list** having four columns (Serial number, the name of the product, quantity and price) for the month of April, 06.

   - Font specifications for Title (Grocery List): 14-point Arial font in bold and italics.
   - The headings of the columns should be in 12-point and bold.
   - The rest of the document should be in 10-point Times New Roman.
   - Leave a gap of 12-points after the title.

2. Create a **telephone directory**.

- The heading should be 16-point Arial Font in bold
- The rest of the document should use 10-point font size
- Other headings should use 10-point Courier New Font.
- The footer should show the page number as well as the date last updated.

3. Design a **time-table form** for your college.
    - The first line should mention the name of the college in 16-point Arial Font and should be bold.
    - The second line should give the course name/teacher‘s name and the department in 14-point Arial.
    - Leave a gap of 12-points.
    - The rest of the document should use 10-point Times New Roman font.
    - The footer should contain your specifications as the designer and date of creation.

4. XYZ Publications plans to release a new book designed as per your syllabus. Design the **first page of the book** as per the given specifications.
    - The title of the book should appear in bold using 20-point Arial font.
    - The name of the author and his qualifications should be in the center of the page in 16-point Arial font.
    - At the bottom of the document should be the name of the publisher and address in 16-point Times New Roman.
    - The details of the offices of the publisher (only location) should appear in the footer.
5. Create the following one page documents.
    a. Compose a note inviting friends to a get-together at your house, Including a list of things to bring with them.
    b. Design a certificate in landscape orientation with a border around the document.
    c. Design a Garage Sale sign.
    d. Make a sign outlining your rules for your bedroom at home, using a numbered list.
6. Create the following documents:
    (a) A newsletter with a headline and 2 columns in portrait orientation, including at least one image surrounded by text.
    (b) Use a newsletter format to promote upcoming projects or events in your classroom or college.
7. Convert following text to a table, using comma as delimiter

    Type the following as shown (do not bold).

    Color, Style, Item
    Blue, A980, Van
    Red, X023, Car
    Green, YL724, Truck
    Name, Age, Sex
    Bob, 23, M
    Linda, 46, F
    Tom, 29, M

9. Enter the following data into a table given on the next page.

Salesperson	Dolls	Trucks	Puzzles

Kennedy, Sally	1327	1423	1193
White, Pete	1421	3863	2934
Pillar, James	5214	3247	5467
York, George	2190	1278	1928
Banks, Jennifer	1201	2528	1203
Atwater, Kelly	4098	3079	2067

Pillar, James	5214	3247	5467
York, George	2190	1278	1928
Banks, Jennifer	1201	2528	1203
Atwater, Kelly	4098	3079	2067

Add a column Region (values: S, N, N, S,S,S) between the Salesperson and Dolls columns to the given table Sort your table data by Region and within Region by Salesperson in ascending order:

In this exercise, you will add a new row to your table, place the word "Total" at the bottom of the Salesperson column, and sum the Dolls, Trucks, and Puzzles columns.

10. Wrapping of text around the image.

11. Following features of menu option must be covered

FILE	Complete menu
EDIT	Complete menu
VIEW	Complete menu
INSERT	Complete menu
FORMAT	Complete menu
TABLE	Complete menu
WINDOW	Complete menu
HELP	Complete menu
TOOLS	All options except online collaboration, Tools on Macro, Templates

MS Excel

1. Enter the Following data in Excel Sheet

REGIONAL SALES PROJECTION

State	Qtr1	Qtr2	Qtr3	QTR4	Qtr Total	Rate Amount

Delhi	2020	2400	2100	3000	15
Punjab	1100	1300	1500	1400	20
U.P.	3000	3200	2600	2800	17
Haryana	1800	2000	2200	2700	15
Rajasthan	2100	2000	1800	2200	20

TOTAL
AVERAGE

(a) Apply Formatting as follow: I.Title

in TIMES NEW ROMAN

ii. Font Size - 14
iii. Remaining text - ARIAL, Font Size -10
iv. State names and Qtr. Heading Bold, Italic with Gray Fill Color.
v. Numbers in two decimal places.
vi. Qtr. Heading in center Alignment.
vii. Apply Border to whole data.

(b) Calculate State and Qtr. Total

(c) Calculate Average for each quarter

(d) Calculate Amount = Rate * Total.

2. Given the following worksheet

	A	**B**	**C**	**D**
1	Roll No.	Name	Marks	Grade
2	1001	Sachin	99	
3	1002	Sehwag	65	
4	1003	Rahul	41	
5	1004	Sourav	89	
6	1005	Har Bhajan	56	

Calculate the grade of these students on the basis of following guidelines:

If Marks	Then Grade
>= 80	A+
>= 60 < 80	A
>= 50 < 60	B
< 50	F

3. Given the following worksheet

	A	**B**	**C**	**D**	**E**	**F**	**G**
1	**Salesman**			**Sales in (Rs.)**			
2	No.	Qtr1	Qtr2	Qtr3	Qtr4	Total	Commission
3	S001	5000	8500	12000	9000		
4	S002	7000	4000	7500	11000		
5	S003	4000	9000	6500	8200		
6	S004	5500	6900	4500	10500		
7	S005	7400	8500	9200	8300		
8	S006	5300	7600	9800	6100		

Calculate the commission earned by the salesmen on the basis of following Candidates:

If Total Sales	**Commission**

< 20000	0% of sales
> 20000 and < 25000	4% of sales
> 25000 and < 30000	5.5% of sales
> 30000 and < 35000	8% of sales
>= 35000	11% of sales

The total sales is sum of sales of all the four quarters.

4. A company XYZ Ltd. pays a monthly salary to its employees which consists of basic salary, allowances & deductions. The details of allowances and deductions are as follows:

Allowances

- HRA Dependent on Basic

  30% of Basic if Basic <=1000

  25% of Basic if Basic>1000 & Basic<=3000

  20% of Basic if Basic >3000

- DA Fixed for all employees, 30% of Basic

- Conveyance Allowance Rs. 50/- if Basic is <=1000

  Rs. 75/- if Basic >1000 & Basic<=2000

  Rs. 100 if Basic >2000

- Entertainment Allowance NIL if Basic

  is <=1000 Rs. 100/- if Basic > 1000

Deductions

- Provident Fund 6% of Basic
- Group Insurance Premium Rs. 40/- if Basic is <=1500

  Rs. 60/- if Basic > 1500 & Basic<=3000

  Rs. 80/- if Basic >3000

Calculate the following:

Gross Salary = Basic + HRA + DA + Conveyance + Entertainment

Total deduction = Provident Fund + Group Insurance Premium

Net Salary = Gross Salary – Total Deduction

5. Create Payment Table for a fixed Principal amount, variable rate of interests and time in the format below:

No. of Instalments	5%	6%	7%	8%	9%
3	XX	XX	XX	XX	XX
4	XX	XX	XX	XX	XX
5	XX	XX	XX	XX	XX
6	XX	XX	XX	XX	XX

6. Use an array formula to calculate Simple Interest for given principal amounts given the rate of Interest and time

Rate of Interest	8%
Time	5 Years
Principal	Simple Interest

1000	?
18000	?
5200	?

7. The following table gives year wise sale figure of five salesmen in Rs.

Salesman	2000	2001	2002	2003
S1	10000	12000	20000	50000
S2	15000	18000	50000	60000
S3	20000	22000	70000	70000
S4	30000	30000	100000	80000
S5	40000	45000	125000	90000

(*a*) Calculate total sale year wise.

(*b*) Calculate the net sale made by each salesman

(*c*) Calculate the maximum sale made by the salesman

(d) Calculate the commission for each salesman under the condition.

(*i*) If total sales >4,00,000 give 5% commission on total sale made by the salesman.

(*ii*) Otherwise give 2% commission.

(*e*) Draw a bar graph representing the sale made by each salesman. (*f*) Draw a pie graph representing the sale made by salesman in 2000.

8. Enter the following data in Excel Sheet

PERSONAL BUDGET FOR FIRST QUARTER

Monthly Income (Net): 1,475

EXPENSES	**JAN**	**FEB**	**MARCH**	**QUARTER**	**QUARTER**
				TOTAL	**AVERAGE**
Rent600.00	600.00	600.00			
Telephone	48.25	43.50	60.00		
Utilities	67.27	110.00	70.00		
Credit Card	200.00	110.00	70.00		
Oil	100.00	150.00	90.00		
AV to Insurance	150.00				
Cable TV	40.75	40.75	40.75		
Monthly Total					

Calculate Quarter total and Quarter average.

(a) Calculate Monthly total.

(b) Surplus = Monthly income - Monthly total.

(c) What would be total surplus if monthly income is 1500.

(d) How much does telephone expense for March differ from quarter average.

(e) Create a 3D column graph for telephone and utilities.

(f) Create a pie chart for monthly expenses.

9. Enter the following data in Excel Sheet

TOTAL REVENUE EARNED FOR SAM’S BOOKSTALL

Publisher name	**1997**	**1998**	**1999**	**2000**	**total**
A	Rs. 1,000.00	Rs. 1100.00	Rs. 1,300.00	Rs. 800.00	
B	Rs. 1,500.00	Rs. 700.00	Rs. 1,000.00	Rs. 2,000.00	
C	Rs. 700.00	Rs. 900.00	Rs. 1,500.00	Rs. 600.00	
D	Rs. 1,200.00	Rs. 500.00	Rs. 200.00	Rs. 1,100.00	
E	Rs 800.00	Rs. 1,000.00	Rs. 3,000.00	Rs. 560.00	

(a) Compute the total revenue earned.

(b) Plot the line chart to compare the revenue of all publisher for 4 years.

(b) Chart Title should be ‗Total Revenue of sam‘s Bookstall (1997-2000)‘

(c) Give appropriate categories and value axis title.

10. Generate 25 random numbers between 0 & 100 and find their sum, average and count. How many no. are in range 50-60.

Vidyasagar University

Curriculum for B.Sc. Honours in Computer Science [Choice Based Credit System]

Semester-II

Sl. No.	Name of the Subject	Nature	Code	Teaching Scheme in hour per week			Credit	Marks
				L	T	P		
C3	**C3T:** Programming in Java	Core Course-3		4	0	0	6	75
	C3P: Programming in Java (Lab)	Core Course-3 [Practical]		0	0	4		
C4	**C4T:** Discrete Structures	Core Course-4		5	1	0	6	75
GE-2	GE-2	GE					4/5	75
	GE-2	GE					2/1	
AEC C-2	Environmental Studies	AECC					4	100
				Total Credits =22				

L=Lecture, T=Tutorial, P=Practical

AECC- Ability Enhancement Compulsory Course: Environmental Studies.

Interdisciplinary/Generic Elective (GE) from other Department
[**Four papers are to be taken and each paper will be of 6 credits**]:
[Papers are to be taken from any of the following discipline **(GE-2 from Mathematics)]: Mathematics/Physics/Chemistry/Statistics/Geology/Electronics/Economics/Zoology/Botany/Physiology**

Semester-II
Core Courses

Core-3

CC-3: Programming in Java   Credits 06

C3T: Programming in Java   Credits 04

(Theory: 60 Lectures)

1. Introduction to Java   (4 Lectures)

Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

2. Arrays, Strings and I/O   (8 Lectures)

Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

3. Object-Oriented Programming Overview   (4 Lectures)

Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

3. Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata
(14 lectures)

Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

4. Exception Handling, Threading, Networking and Database Connectivity
(15 Lectures)

Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

5. Applets and Event Handling   (15 Lectures)

Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts. Overview of servlets.

Reference Books

1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.
2. James Gosling, Bill Joy, Guy L Steele Jr, Gilad Bracha, Alex Buckley "The Java Language Specification, Java SE 8 Edition (Java Series)", Published by Addison Wesley, 2014.
3. Joshua Bloch, "Effective Java" 2nd Edition, Publisher: Addison-Wesley, 2008.
4. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 1 ,9th Edition, Printice Hall.2012
5. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features)", 9th Edition, Printice Hall.2013
6. Bruce Eckel, "Thinking in Java", 3rd Edition, PHI, 2002.
7. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.2009.
8. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.
9. "Head First Java", Orielly Media Inc. 2nd Edition, 2005.
10. David J. Eck, "Introduction to Programming Using Java", Published by CreateSpace Independent Publishing Platform, 2009.
11. John R. Hubbard, "Programming with JAVA", Schaum's Series, 2nd Edition, 2004.

C3P: Programming in Java (Lab) Credits 02

(Practical: 60 Lectures)

1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. To learn use of lenth in case of a two dimensional array
5. To convert a decimal to binary number
6. To check if a number is prime or not, by taking the number as input from the keyboard
7. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
8. Write a program that show working of different functions of String and String Buffer classs like set Charat (set Length (), append (), insert (), concat ()and equals ().
9. Write a program to create a class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
10. Modify the class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
11. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)
12. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword

13. Write a program to show the use of static functions and to pass variable length arguments in a function.
14. Write a program to demonstrate the concept of boxing and unboxing.
15. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
16. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacii series is given in a different file belonging to the same package.
17. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. Write a program that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
20. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
21. Write a program to demonstrate priorities among multiple threads.
22. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
23. Write a program to create URL object, create a URL Connection using the open Connection () method and then use it examine the different components of the URL and content.
24. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
25. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet‘s window.
26. Write a program to get the URL/location of code (i.e. java code) and document (i.e. html file).
27. Write a program to demonstrate different mouse handling events like mouse Clicked (), mouse Entered (), mouse Exited (), mouse Pressed, mouse Released () and mouse Dragged ().
28. Write a program to demonstrate different keyboard handling events.
29. Write a program to generate a window without an applet window using main () function.
30. Write a program to demonstrate the use of push buttons.

Core-4

CC-4: Discrete Structures Credits 06

C4T: Discrete Structures Credits 04

(Theory: 60 Lectures)

1. Introduction: (15 Lectures)

Sets - finite and Infinite sets, uncountably Infinite Sets; functions, relations, Properties of Binary Relations, Closure, Partial Ordering Relations; counting - Pigeonhole Principle, Permutation and Combination; Mathematical Induction, Principle of Inclusion and Exclusion.

2. Growth of Functions: **(8 Lectures)**

Asymptotic Notations, Summation formulas and properties, Bounding Summations, approximation by Integrals

3. Recurrences: **(10 Lectures)**

Recurrence Relations, generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Recurrence Trees, Master Theorem

4.Graph Theory **(15 Lectures)**

Basic Terminology, Models and Types, multigraphs and weighted graphs, Graph Representation, Graph Isomorphism, Connectivity, Euler and Hamiltonian Paths and Circuits, Planar Graphs, Graph Coloring, Trees, Basic Terminology and properties of Trees, Introduction to Spanning Trees

5. Prepositional Logic **(12 Lectures)**

Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory

Recommended Books:

1. C.L. Liu , D.P. Mahapatra, Elements of Discrete mathematics, 2nd Edition , Tata McGraw Hill, 1985,
2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition ,McGraw Hill 2006
3. T.H. Coremen, C.E. Leiserson, R. L. Rivest, Introduction to algorithms, 3rd edition Prentice Hall on India, 2009
4. M. O. Albertson and J. P. Hutchinson, Discrete Mathematics with Algorithms , John wiley Publication, 1988
5. J. L. Hein, Discrete Structures, Logic, and Computability, 3rd Edition, Jones and Bartlett Publishers, 2009
6. D.J. Hunter, Essentials of Discrete Mathematics, Jones and Bartlett Publishers, 2008

Generic Elective Syllabus

GE-2 (Interdisciplinary for other department)

GE -2 : Introduction to Database System Credits 06

GE 2 T : Introduction to Database System Credits 04

(Theory: 60 lectures)

Database: Introduction to database, relational data model, DBMS architecture, data independence, DBA, database users, end users, front end tools **14L**

E-R Modeling: Entity types, entity set, attribute and key, relationships, relation types, E- R diagrams, database design using ER diagrams **14L**

Relational Data Model: Relational model concepts, relational constraints, primary and foreign key, normalization: 1NF, 2NF, 3NF **14L**

Structured Query Language: SQL queries, create a database table, create relationships between database tables, modify and manage tables, queries, forms, reports, modify, filter and view data. **18L**

Reference Books :

1. P. Rob, C. Coronel, Database System Concepts by, Cengage Learning India, 2008
2. R. Elmsasri,S. Navathe Fundamentals of Database Systems, Pearson Education, Fifth Edition, 2007
   MySQL : Reference Manual

GE2 P : Introduction to Database System (Lab) Credits 02

(Practical: 60 lectures)

1) Create a database having two tables with the specified fields, to computerize a library system of a Delhi University College.
Library Books (Accession number, Title, Author, Department, Purchase Date, Price) Issued Books (Accession number, Borrower)

a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
b) Delete the record of book titled ―Database System Concepts‖.
c) Change the Department of the book titled ―Discrete Maths‖ to ―CS‖.
d) List all books that belong to ―CS‖ department.
e) List all books that belong to ―CS‖ department and are written by author ―Navathe‖.
f) List all computer (Department=‖CS‖) that have been issued.
g) List all books which have a price less than 500 or purchased between ―01/01/1999‖ and ―01/01/2004‖.

2) Create a database having three tables to store the details of students of Computer Department in your college.

Personal information about Student (College roll number, Name of student, Date of birth, Address, Marks(rounded off to whole number) in percentage at 10 + 2, Phone number) Paper Details (Paper code, Name of the Paper)

Student's Academic and Attendance details (College roll number, Paper code, Attendance, Marks in home examination).

a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
b) Design a query that will return the records (from the second table) along with the name of student from the first table, related to students who have more than 75% attendance and more than 60% marks in paper 2.

c) List all students who live in —Delhi‖ and have marks greater than 60 in paper 1.
d) Find the total attendance and total marks obtained by each student.
e) List the name of student who has got the highest marks in paper 2.

3) Create the following tables and answer the queries given below:
Customer (Cust ID, email, Name, Phone, Referrer ID)
Bicycle (Bicycle ID, Date Purchased, Color, Cust ID,
Model No) Bicycle Model (Model No, Manufacturer,
Style) Service (Start Date, Bicycle ID, End Date)

a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
b) List all the customers who have the bicycles manufactured by manufacturer —Honda‖.
c) List the bicycles purchased by the customers who have been referred by customer —C1‖.
d) List the manufacturer of red colored bicycles.
e) List the models of the bicycles given for service.

4) Create the following tables, enter at least 5 records in each table and answer the queries given below.

EMPLOYEE (Person_Name, Street, City)
WORKS (Person_Name, Company_Name,
Salary) COMPANY (Company_Name, City)
MANAGES (Person_Name, Manager_Name)

a) Identify primary and foreign keys.
b) Alter table employee, add a column —email‖ of type varchar (20).
c) Find the name of all managers who work for both Samba Bank and NCB Bank.
d) Find the names, street address and cities of residence and salary of all employees who work for —Samba Bank‖ and earn more than $10,000.
e) Find the names of all employees who live in the same city as the company for which they work.
f) Find the highest salary, lowest salary and average salary paid by each company.
g) Find the sum of salary and number of employees in each company.
h) Find the name of the company that pays highest salary.

5) Create the following tables, enter at least 5 records in each table and answer the queries given below.

Suppliers (SNo, Sname, Status, SCity)
Parts (PNo, Pname, Colour, Weight,
City) Project (JNo, Jname, Jcity)
Shipment (Sno, Pno, Jno, Qunatity)

a) Identify primary and foreign keys.
b) Get supplier numbers for suppliers in Paris with status>20.
c) Get suppliers details for suppliers who supply part P2. Display the supplier list in increasing order of supplier numbers.
d) Get suppliers names for suppliers who do not supply part P2.
e) For each shipment get full shipment details, including total shipment weights.
f) Get all the shipments where the quantity is in the range 300 to 750 inclusive.

g) Get part nos. for parts that either weigh more than 16 pounds or are supplied by suppliers S2, or both.

h) Get the names of cities that store more than five red parts.

i) Get full details of parts supplied by a supplier in London.

j) Get part numbers for part supplied by a supplier in London to a project in London.

k) Get the total number of project supplied by a supplier (say, S1).

l) Get the total quantity of a part (say, P1) supplied by a supplier (say, S1).

Vidyasagar University

Curriculum for B.Sc (Honours) in Computer Science
[Choice Based Credit System]

Semester-III

Course	**Course Code**	**Name of the Subjects**	**Course Type/ Nature**	**Teaching Scheme in hour per week**			**Credit**	**Marks**
				L	**T**	**P**		
CC-5		**C5T: :** Data Structures	Core Course - 5	4	0	0	6	75
		C5P: Data Structures Lab		0	0	4		
CC-6		**C6T:** Operating Systems	Core Course - 6	4	0	0	6	75
		C6P: Operating Systems Lab		0	0	4		
CC-7		**C7T:** Computer Networks	Core Course - 7	4	0	0	6	75
		C7P: Computer Networks Lab		0	0	4		
GE-3		**TBD**	Generic Elective -3				4/5	75
							2/1	
SEC-1		**SEC-1T:** Android Programming **SEC-1P:** Software Lab Based on Android Programming **Or** **SEC-1T:** Programming in MATLAB **SEC1P:** Software Lab Based on MATLAB	Skill Enhancement Course-1	1	0	2	2	50
Semester Total							**26**	**350**

L=Lecture, **T=** Tutorial, **P=**Practical, **CC =** Core Course, **GE**= Generic Elective, **SEC** = Skill Enhancement Course, **TBD =** to be decided

Generic Elective (GE) (Interdisciplinary) from other Department [**Four papers are to be taken and each paper will be of 6 credits**]: Papers are to be taken from any of the following discipline: **Mathematics/Physics/Chemistry/Statistics/Geology/Electronics/Economics/ Zoology/ Botany/Physiology**

Modalities of selection of Generic Electives (GE): A student shall have to choose **04** Generic Elective (GE1 to GE4) strictly from **02** subjects / disciplines of choice taking exactly **02** courses from each subjects of disciplines. Such a student shall have to study the curriculum of Generic Elective (GE) of a subject or discipline specified for the relevant semester.

Semester-III

Core Course (CC)

CC-5: Data Structures Credits 06

C5T: Data Structures Credits 04

Theory: 60 Lectures

1. Arrays (5 Lectures)

Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation)

2. Stacks (5 Lectures)

Implementing single / multiple stack/s in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack; Limitations of Array representation of stack

3. **Linked Lists (10 Lectures)**

Singly, Doubly and Circular Lists (Array and Linked representation); Normal and Circular representation of Stack in Lists; Self Organizing Lists; Skip Lists

4. Queues (5 Lectures)

Array and Linked representation of Queue, De-queue, Priority Queues

5. **Recursion (5 lectures)**

Developing Recursive Definition of Simple Problems and their implementation; Advantages and Limitations of Recursion; Understanding what goes behind Recursion (Internal Stack Implementation)

6. **Trees (20 Lectures)**

Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL Trees). Tree traversal techniques.

7. **Searching and Sorting (5 Lectures)**

Linear Search, Binary Search, Comparison of Linear and Binary Search, Selection Sort, Insertion Sort, Bubble Sort, Quick Sort, Comparison of Sorting Techniques

8. **Hashing (5 Lectures)**

Introduction to Hashing, Efficiency of Rehash Methods, Resolving collision by Open Addressing, Coalesced Hashing, Separate Chaining, Dynamic and Extendible Hashing.

Reference Books:

1. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
2. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
3. Aaron M. Tenenbaum, Moshe J. Augenstein, Yedidyah Langsam, "Data Structures Using C and C++:, Second edition, PHI, 2009.
4. Robert L. Kruse, "Data Structures and Program Design in C++", Pearson, 1999.
5. D.S Malik, Data Structure using C++, Second edition, Cengage Learning, 2010.
6. Mark Allen Weiss, ″Data Structures and Algorithms Analysis in Java", Pearson Education, 3rd edition, 2011
7. Aaron M. Tenenbaum, Moshe J. Augenstein, Yedidyah Langsam, "Data Structures Using Java, 2003.
8. Robert Lafore, "Data Structures and Algorithms in Java, 2/E", Pearson/ Macmillan Computer Pub, 2003
9. John Hubbard, "Data Structures with JAVA", McGraw Hill Education (India) Private Limited; 2 edition, 2009
10. Goodrich, M. and Tamassia, R. "Data Structures and Algorithms Analysis in Java", 4th Edition, Wiley, 2013
11. Herbert Schildt, ″Java The Complete Reference (English) 9th Edition Paperback", Tata McGraw Hill, 2014.
12. D. S. Malik, P.S. Nair, "Data Structures Using Java", Course Technology, 2003.

C5P: Data Structures Lab **Credits 02**

Practical: 60 Lectures

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
6. Perform Stack operations using Linked List implementation.
7. Perform Stack operations using Array implementation. Use Templates.
8. Perform Queues operations using Circular Array implementation. Use Templates.
9. Create and perform different operations on Double-ended Queues using Linked List implementation.
10. WAP to scan a polynomial using linked list and add two polynomial.
11. WAP to calculate factorial and to compute the factors of a given no. (i) using recursion, (ii) using iteration
12. (ii) WAP to display fibonacci series (i)using recursion, (ii) using iteration
13. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion

14. WAP to create a Binary Search Tree and include following operations in tree:

    (a) Insertion (Recursive and Iterative Implementation)
    (b) Deletion by copying
    (c) Deletion by Merging
    (d) Search a no. in BST
    (e) Display its preorder, postorder and inorder traversals Recursively
    (f) Display its preorder, postorder and inorder traversals Iteratively
    (g) Display its level-by-level traversals
    (h) Count the non-leaf nodes and leaf nodes
    (i) Display height of tree
    (j) Create a mirror image of tree
    (k) Check whether two BSTs are equal or not

15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
16. WAP to reverse the order of the elements in the stack using additional stack.
17. WAP to reverse the order of the elements in the stack using additional Queue.
18. WAP to implement Diagonal Matrix using one-dimensional array.
19. WAP to implement Lower Triangular Matrix using one-dimensional array.
20. WAP to implement Upper Triangular Matrix using one-dimensional array.
21. WAP to implement Symmetric Matrix using one-dimensional array.
22. WAP to create a Threaded Binary Tree as per in order traversal, and implement operations like finding the successor / predecessor of an element, insert an element, in order traversal.
23. WAP to implement various operations on AVL Tree.

CC-6: Operating Systems Credits 06

C6T: Operating Systems Credits 04

Theory: 60 Lectures

1. Introduction (10 Lectures)

Basic OS functions, resource abstraction, types of operating systems–multiprogramming systems, batch systems , time sharing systems; operating systems for personal computers & workstations, process control & real time systems.

Case study on Linux system (6 Lectures)

- Cloud computing (3 lectures)
- Linux evolution and Linux distros (2 lectures)
- Linux file system (1 lecture)

2. Operating System Organization (6 Lectures)

Processor and user modes, kernels, system calls and system programs.

3. Process Management (16 Lectures)

System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-pre-emptive and pre-emptive scheduling algorithms; concurrent processes, critical section, semaphores, methods for inter-process communication; deadlocks.

4. Memory Management (10 Lectures)

Physical and virtual address space; memory allocation strategies – fixed and variable partitions, paging, segmentation, virtual memory

5. File and I/O Management (8 Lectures)

Directory structure, file operations, file allocation methods, device management.

6. Protection and Security (4 Lectures)

Policy mechanism, Authentication, Internal access Authorization.

Recommended Books:

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education 1997.
4. W. Stallings, Operating Systems, Internals & Design Principles , 5th Edition, Prentice Hall of India. 2008.
5. M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.

C6P: Operating Systems Lab Credits 02

Practical: 60 Lectures

C/ C++ programs

1. Write a program (using *fork ()* and/or *exec ()* commands) where parent and child execute:
   a) same program, same code.
   b) same program, different code.
   c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behaviour of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behaviour of Linux kernel including information on configured memory, amount of free and used memory (memory information).
4. Write a program to print file details including owner access permissions, file access time, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write program to implement FCFS scheduling algorithm.
7. Write program to implement Round Robin scheduling algorithm.
8. Write program to implement SJF scheduling algorithm.
9. Write program to calculate sum of n numbers using *thread* library.
10. Write a program to implement first-fit, best-fit and worst-fit allocation strategies

CC-7: Computer Networks Credits 06

C7T: Computer Networks Credits 04

Theory: 60 Lectures

1. Introduction to Computer Networks (8 Lectures)
Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite.

2. Data Communication Fundamentals and Techniques (10 Lectures)

Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation-; multiplexing techniques- FDM, TDM; transmission media.

3. Networks Switching Techniques and Access mechanisms (10 Lectures)
Circuit switching; packets witching- connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.

4. Data Link Layer Functions and Protocol (10 Lectures)
Error detection and error correction techniques; data-link control- framing and flow control; error recovery protocols- stop and wait ARQ, go-back-n ARQ; Point to Point Protocol on Internet.

5. Multiple Access Protocol and Networks (5 Lectures)
CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks- repeaters, hubs, switches, bridges, router and gateways;

6. Networks Layer Functions and Protocols (6 Lectures)
Routing; routing algorithms; network layer protocol of Internet- IP protocol, Internet control protocols.

7. Transport Layer Functions and Protocols (6 Lectures)
Transport services- error and flow control, Connection establishment and release – three way handshake;

8. Overview of Application layer protocol (5 Lectures)
Overview of DNS protocol; overview of WWW &HTTP protocol.

Reference Books

1. B. A. Forouzan: Data Communications and Networking, Fourth edition, THM, 2007.
2. A. S. Tanenbaum: Computer Networks**,** Fourth edition**,** PHI, 2002.

C7P: Computer Networks Lab **Credits 02**
Practical: 60 Lectures

1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm
6. Simulate and implement Dijkstra algorithm for shortest path routing.
7. Experiments for capturing and analyzing data packets using Wire Shark.
    - Experiments on filtering packets
    - Experiments on inspecting packets

Skill Enhancement Course (SEC)

SEC-1: Android Programming Credits 02

SEC-1T: Android Programming Credits 01

Introduction: History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.

(2L)

Overview of object oriented programming using Java: OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

(4L)

Development Tools: Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project – Hello Word, run on emulator, Deploy it on USB-connected Android device.

(5L)

User Interface Architecture: Application context, intents, Activity life cycle, multiple screen sizes.

(2L)

User Interface Design: Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners(Combo boxes),Images, Menu, Dialog.

(2L)

Database: Understanding of SQLite database, connecting with the database.

(2L)

Book Recommended:

1. Android application development for java programmers. By James C. Sheusi. Publisher: Cengage Learning, 2013.

Online Reading / Supporting Material:

1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://docs.oracle.com/javase/tutorial/index.htm (Available in the form of free downloadable ebooks also).
5. http://developer.android.com/guide/components/activities.html
6. http://developer.android.com/guide/components/fundamentals.html
7. http://developer.android.com/guide/components/intents-filters.html.
8. http://developer.android.com/training/multiscreen/screensizes.html
9. http://developer.android.com/guide/topics/ui/controls.html
10. http://developer.android.com/guide/topics/ui/declaring-layout.html
11. http://developer.android.com/training/basics/data-storage/databases.html

SEC-1P: Software Lab Based on Android Programming Credits 01

1. Create —Hello World‖ application. That will display —Hello World‖ in the middle of the screen in the emulator. Also display —Hello World‖ in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

OR

SEC-1: Programming in MATLAB Credits 02

SEC-1T: Programming in MATLAB Credits 01

Unit I- Introduction to Programming: Components of a computer, working with numbers, Machine code, Software hierarchy

(2L)

Unit II- Programming Environment: MATLAB Windows, A First Program, Expressions, Constants, Variables and assignment statement, Arrays

(3L)

Unit III- Graph Plots: Basic plotting, Built in functions, Generating waveforms, Sound replay, load and save.

(2L)

Unit IV- Procedures and Functions: Arguments and return values, M-files, Formatted console input-output, String handling.

(3L)

Unit V-Control Statements: Conditional statements: If, Else, Else-if, Repetition statements: While, for loop.

(2L)

Unit VI- Manipulating Text: Writing to a text file, Reading from a text file, Randomising and sorting a list, searching a list.

(2L)

Unit VII- GUI Interface: Attaching buttons to actions, Getting Input, Setting Output.

(2L)

Recommended Books:

1. MATLAB: An Introduction with Applications, by Amos Gilat, 2nd edition, Wiley, 2004,
2. C.B. Moler, Numerical Computing with MATLAB, SIAM, 2004.

SEC1P: Software Lab Based on MatLab **Credits 01**

1. Write a program to assign the following expressions to a variable A and then to print out the value of A.
    a) (3+4)/(5+6)
    b) $2\pi^2$
    c) $\sqrt{2}$
    d) $(0.0000123 + 5.67 \times\ 10^{-3}) \times 0.4567 \times 10^{-4}$

2. Celsius temperatures can be converted to Fahrenheit by multiplying by 9, dividing by 5, and adding 32. Assign a variable called C the value 37, and implement this formula to assign a variable F the Fahrenheit equivalent of 37 Celsius.

3. Set up a vector called N with five elements having the values: 1, 2, 3, 4, 5. Using N, create assignment statements for a vector X which will result in X having these values:

    a. 2, 4, 6, 8, 10
    b. 1/2, 1, 3/2, 2, 5/2
    c. 1, 1/2, 1/3, 1/4, 1/5
    d. 1, 1/4, 1/9, 1/16, 1/25

4. A supermarket conveyor belt holds an array of groceries. The price of each product (in pounds) is [0.6, 1.2 ,0.5, 1.3] ; while the numbers of each product are [3, 2 ,1 ,5]. Use MATLAB to calculate the total bill.
5. The sortrows(x) function will sort a vector or matrix X into increasing row order. Use this function to sort a list of names into alphabetical order.
6. The ―identity‖ matrix is a square matrix that has ones on the diagonal and zeros elsewhere. You can generate one with the eye() function in MATLAB. Use MATLAB to find a matrix B, such that when multiplied by matrix A=[1 2; -1 0] the identity matrix I=[1 0; 0 1] is generated. That is A*B=I.
7. Create an array of N numbers. Now find a single MATLAB statement that picks out from that array the 1,4,9,16,…,√Nth entries, i.e. those numbers which have indices that are square numbers.
8. Draw a graph that joins the points (0,1), (4,3), (2,0) and (5,-2).
9. The seeds on a sunflower are distributed according to the formula below. Plot a small circle at each of the first 1000 co-ordinates:

$$r_{n=\sqrt{n}}$$

$$\theta_n = \frac{137.51}{180}\pi n$$

10. Calculate 10 approximate points from the function y=2x by using the formulae:

    i. xn = n
    ii. yn = 2n + rand - 0.5

    Fit a line of best fit to these points using the function polyfit() with degree=1, and generate co-ordinates from the line of best fit using polyval(). Use the on-line help to find out how to use these functions. Plot the raw data and the line of best fit.
11. Calculate and replay 1 second of a sinewave at 500Hz with a sampling rate of 11025Hz. Save the sound to a file called "ex35.wav". Plot the first 100 samples.

12. Calculate and replay a 2 second chirp. That is, a sinusoid that steadily increases in frequency with time, from say 250Hz at the start to 1000Hz at the end.
13. Build a square wave by adding together 10 odd harmonics: 1f, 3f, 5f, etc. The amplitude of the nth harmonic should be 1/n. Display a graph of one cycle of the result superimposed on the individual harmonics.
14. Write a function called FtoC (ftoc.m) to convert Fahrenheit temperatures into Celsius. Make sure the program has a title comment and a help page. Test from the command window with:

    i. FtoC(96)
    ii. lookfor Fahrenheit
    iii. help FtoC

15. Write a program to input 2 strings from the user and to print out (i) the concatenation of the two strings with a space between them, (ii) a line of asterisks the same length as the concatenated strings, and (iii) the reversed concatenation. For example:

    i. Enter string 1: Mark
    ii. Enter string 2: Huckvale
    iii. Mark Huckvale
    iv. *************
    v. elavkcuH kraM

Generic Elective Syllabus

GE-3 [Interdisciplinary for other department]

GE-3: Introduction to Programming Credits 06

GE3P: Introduction to Programming Credits 04

Theory: 60 lectures

Introduction to C and C++ (5 Lectures)

History of C and C++, Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C++.

Data Types, Variables, Constants, Operators and Basic I/O (10 Lectures)

Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf(), cin, cout), Using Basic Header Files (stdio.h, iostream.h, conio.hetc).

Expressions, Conditional Statements and Iterative Statements (10 Lectures)

Simple Expressions in C++ (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Functions and Arrays (10 Lectures)

Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.

Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays

Derived Data Types (Structures and Unions) (5 Lectures)

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

File I/O, Preprocessor Directives (8 Lectures)

Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros

Using Classes in C++ **(8 Lectures)**

Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables &Functions, Objects as parameters, specifying the Protected and Private Access, Copy Constructors, Overview of Template classes and their use.

Inheritance and Polymorphism **(4 Lectures)**

Introduction to Inheritance and Polymorphism

Reference Books:

1. Herbtz Schildt, "C++: The Complete Reference", Fourth Edition, McGraw Hill.2003
2. Bjarne Stroustrup, "The C++ Programming Language", 4th Edition, Addison-Wesley , 2013.
3. Bjarne Stroustroup, "Programming -- Principles and Practice using C++", 2nd Edition, Addison-Wesley 2014.
4. E Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill Education, 2008.
5. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice Hall, 2011.
6. John R. Hubbard, "Programming with C++", Schaum's Series, 2nd Edition, 2000.
7. Andrew Koeni, Barbara, E. Moo, "Accelerated C++", Published by Addison-Wesley , 2000.
8. Scott Meyers, "Effective C++", 3rd Edition, Published by Addison-Wesley, 2005.
9. Harry, H. Chaudhary, "Head First C++ Programming: The Definitive Beginner's Guide", First Create space Inc, O-D Publishing, LLC USA.2014
10. Walter Savitch, "Problem Solving with C++", Pearson Education, 2007.
11. Stanley B. Lippman, JoseeLajoie, Barbara E. Moo, "C++ Primer", Published by Addison-Wesley, 5th Edition, 2012

GE3P: Introduction to c/c++ Programming Lab **Credits 02**

Practical: 60 lectures

1. Write a program to find greatest of three numbers.
2. Write a program to find gross salary of a person
3. Write a program to find grade of a student given his marks.
4. Write a program to find divisor or factorial of a given number.
5. Write a program to print first ten natural numbers.
6. Write a program to print first ten even and odd numbers.
7. Write a program to find grade of a list of students given their marks.

8. Create Matrix class. Write a menu-driven program to perform following Matrix operations (2-D array implementation):
   a) Sum b) Difference c) Product d) Transpose