Guidelines for B.Sc. (H) Computer Science Semester II

These are only guidelines and in no case to be treated as syllabus. The guidelines may be updated from time to time. Click here for changes that were made in September, 2008.

December 5, 2006

201 Data Structures

Chapter	Reference
Ch. 4, 5, 7, 10	[1]
Overview of Ch. 11 and 12	[1]
Ch. 13, 14, 15, 17, 18	[1]
Overview of Ch. 19	[1]
Ch. 3	[2]
Ch. 4	[2]
Ch. 5 (5.1 to 5.10)	[2]
Ch. 6 (6.1 to 6.6)	[2]
Ch 4 (4.1 to 4.6)	[3]

References:

[1] H. Schildt, The Complete Reference C++, Third edition, Tata McGraw Hill

[2] Adam Drozdek, Data Structures and algorithm in C++, Second Edition, Vikas Publications

[3] Sartaj Sahni, Data Structures, Algorithms and applications in C++, McGraw Hill

202 Computer Systems Architecture

Chapter	Title	Reference
3 (3.3 to 3.5) upto bus structure skip PCI commands	A top level view of computer function and interconnection	[1]
Appendix 3A	Timing diagram	[1]
4 (4.1 to 4.3)	Cache memory	[1]
Appendix 4A	Performance characteristics of two level memories	[1]
5 (complete)	Basic computer organization & design	[2]
6 (6.1 to 6.3 till pg. 181)	Programming the basic computer	[2]
7	Micro-programmed control organization	[2]
7 (Complete) (except pg. 213 to 215, Fig. 7.11 on pg. 217)	Input/Output organization	[1]
8 (8.8 only RISC & CISC introduction)	Central Processing unit	[2]
9 (9.3 onwards)	Computer arithmetic	[1]
11 (11.1)	Instruction sets: Addressing modes & formats	[1]
12 (12.4 Instruction pipelining)	CPU structure and function	[1]

References:

[1] William Stallings. Computer Organization and Architecture, Sixth Edition, Pearson Education Asia,.

[2] M. Morris Mano. Computer System Architecture, 3rd Edition, PHI

203 Calculus II

Chapter	Section	Class in Lab	Class in classroom
4[1]	4.4	2	
	4.5	1	2
	4.6	1	2
	4.7	1	1
5[1]	5.2	1	On Theory class for all these sections
	5.3	1	
	5.4	1	
	5.6	1	
11[1]	11.1	1	2
13[1]	13.1	1	
	13.2	2	
	13.3	2	1
	13.4	1	
	13.5	1	
	13.6	2	
	13.7		1
10[2]	6 (Pg.635-654)		
12[3]	12.3	1	1
	12.4		1
	12.5		
	12.6		2 Classes for (12.6-12.8)
	12.7		
	12.8		
	12.9		
13[3]	13.1	1	
	13.2		1 (up to Pg 717)
	13.3		1 class for (13.3-13.4)
	13.4		
14[3]	14.2		1
	14.4		1
15[3]	15.1		1
	15.3		1
	15.4		2
10[2]	10.6	2	
10[3]	Supplementary Reading		

References:

[1] G.B. Thomas and R.L. Finney, Calculus and Analytic Geometry, Addison Wesley

[2] D.Hughes-Hallet, A.M. Gleason, Calculus, John Wiley & Sons

[3] E. Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons

Minimal set of exercises to be done:

Exercises	Reference
Page 321 (Q29, Q32, Q35, Q37, Q39, Q42, Q46, Q53, Q83)	[1]
Page 331 (Q25, Q33, Q43, Q39, Q41)	[1]
Page 338 (Q49, Q55, Q57, Q51, Q54, Q60)	[1]
Page 377 (Q3, Q5, Q7)	[1]
Page 385 (Q7, Q17, Q21, Q37)	[1]
Page 392-393 (Q9, Q27, Q35)	[1]
Page 405 (Q3, Q11)	[1]
Page 865 (Q3, Q5, Q7, Q11, Q17, Q21, Q25, Q27, Q31, Q35)	[1]
Page 1010 (Q3, Q9, Q11, Q13, Q17, Q19, Q21, Q25, Q29, Q39, Q41, Q43, Q51, Q53)	[1]
Page 1018 (Q13, Q15, Q17)	[1]
Page 1024 (Q3, Q7, Q11, Q22, Q27, Q31)	[1]
Page 1032 (Q9, Q15, Q19, Q29, Q23, Q39)	[1]
Page 1037 (Q7)	[1]
Page 1044 (Q3, Q7, Q17, Q25)	[1]
Page 1054 (Q11, Q21)	[1]
Page 647 (Q1-4, Q6, Q7, Q15)	[2]
Page 668 (Q1, Q7, Q8, Q13)	[3]
Page 673 (Q5, Q11, Q14, Q25, Q17, Q23)	[3]
Page 678 (Q4, Q7, Q13)	[3]
Page 682 (Q17, Q15, Q13, Q19)	[3]
Page 687 (Q9, Q11, Q17, Q15)	[3]
Page 691 (Q7, Q13, Q19, Q27)	[3]
Page 698 (Q7, Q9, Q11, Q15)	[3]
Page 711 (Q1, Q7, Q11, Q19, Q23)	[3]
Page 720 (Q1, Q5, Q17, Q21, Q11, Q15)	[3]
Page 724-725 (Q3, Q11, Q17)	[3]
Page 729 (Q1, Q5, Q9, Q17)	[3]
Page 745 (Q1, Q3, Q11, Q15)	[3]
Page 757-758 (Q2, Q3, Q5)	[3]
Page 759 (Q17, Q19, Q23, Q26)	[3]
Page 775-776 (Q1, Q3, Q4, Q11, Q15, Q17)	[3]
Page 786 (Q1, Q3, Q5, Q7, Q9, Q13, Q15, Q17, Q18, Q20)	[3]
Page 793-794 (Q1, Q3, Q5, Q9, Q13, Q17, Q21)	[3]

204 – Probability

Reference [1]

Ch 1 Complete

Ch 2 §2.1 to §2.7

Ch 3 § 3.1 to § 3.6.2

Ch 4 §4.1, § 4.2

§ 4.3 (proposition 4.1 without proof)
up to example 4.15
evaluation of n ! (Remarks on page 199)

§ 4.4 (Theorem 4.1 statement only)
up to example 4.19 § 4.5.1

Ch 11 § 11.1,

§11.2 up to § 11.2.2

(Proposition 11.1 & 11.2 without proof)

§ 11.3.1, § 11.4 (up to example 11.9)

Reference [2]

Ch 15 § 15.1

Minimal Set of Exercises to be done:

Reference [1]

Chapter 1	# 5, 13, 24, 32, 41, 45, 47, 48
Chapter 2	# 1, 9, 15, 20, 28, 34, 36, 37, 40, 42, 46, 54, 70, 77
Chapter 3	# 3, 7, 11, 13, 15, 16, 25, 40, 46 (also $Cov[Y, E(Y \mid X)] = Var[E(Y \mid X)]$), 53, 63, 77
Chapter 4	# 1, 5, 14, 21, 30, 34, 47, 58
Chapter 11	# 1, 2, 7, 12, 15, 33

Reference [2]

§ 15.1 (p458-467) Problems 1-6 p.491

REFERENCES

[1] Introduction to Probability Models, by Sheldon Ross, 8th Ed, Elsevier India limited 2004.

[2] Introduction to Applied Probability by I. F. Blake (John Wiley & Sons)

205 (i)- Analog Electronics

Chapter	Reference
Ch 1	[1]
Ch 2 (2.1 to 2.3, 2.6 to 2.8, 2.11)	[1]
Ch 3 (3.1-3.2)	[1]
Ch 3	[3]
Ch 4 (4.4, 4.5, 4.8 to 4.10)	[2]
Ch 8 (8.2 to 8.6)	[2]
Ch 9 (9.2 to 9.5)	[2]
Ch 16 (16.1-16.5, 16.9, 16.11-16.15)	[2]
Ch 18 (18.1-18.3, 18.5-18.8)	[2]
Ch 3	[3]
Ch 14 (14.1-14.9)	[3]
Ch 15 (15.1-15.4)	[3]
Ch 16 (16.1-16.5)	[2]

References:

[1] Jacob Millman & Arvind Grabel, Microelectronics, McGraw Hill

[2] Jacob Millman & Christos C. Halkias, Integrated Electronics, Tata McGraw-Hill

[3] Allen Mottershead, Electronics Devices and Circuits, Prentice-Hall India.

205 (ii)- Financial Management

Chapter	Reference
Ch 1	[1]
Ch 2 (2.1)	[1]
Ch 4, 6, 9, 12, 17-20, 24-27	[1]

Reference:

[1] P. Chandra, Financial Management, Tata McGraw-Hill

205 (iii)- Logic

Topic	Reference	Chapter
Nature of Logic, sentence and proposition, Logical form, truth and validity	[1]	1
Definition: Nature & classification	[1]	5
Nature &types of classification	[2]	12
Traditional logic: Types of Aristotelian Categorical propositions	[1]	7
Connotation and denotation of terms, distribution of terms	[1]	1
Theory of square of opposition of categorical Propositions, the problem of existential import	[1]	7, 8, 9
Laws of conversion, obversion, and contraposition of categorical propositions	[1]	7, 8, 9
Syllogism: models and figures of syllogism, fallacies of syllogistic arguments.	[1]	7, 8, 9
Syllogistics and method of antilogism	[2]	Pg 91-94
Laws of thought	[2]	9
Informal fallacies	[1]	6
Philosophy of Logic: Fluidity of Logic, Philosophic problem of induction	[3]	6, 8, 9

References:

[1] I M Copi, C Cohen, Introduction to Logic, Pearson Education

[2] M R Cohen, E Nagel, An Introduction to Logic and Scientific Method, Allied Publishers Limited

[3] P F Strawson, Introduction to Logical Theory, B I Publications

206- Software lab based on 201 (C++ and Data Structures)

1. WAP to perform following operations on integer entered as Command Line Argument:
   i) To check if a number is prime or not.
   ii) To print the sum and products of the number's digits
   iii) To reverse the number

2. Write a function that checks if a string is Palindrome or not. Use this function to check user-entered strings.
3. WAP to perform following on user entered array:
   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) Sort the array
   vi) Remove the duplicates from the ordered array
   vii) Compact the array
   viii) Print the array in reverse order

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

4. 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
5. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
6. WAP to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
7. WAP that prints a table indicating the no. of occurrences of each letter of alphabet in the text entered as command line arguments.
8. Write a macro that calculates the area of a circle. WAP to use it.
9. Write a macro that swaps two numbers. WAP to use it.
10. WAP to perform following operations on strings:
    a) Show address of each character in string
    b) Concatenate 2 string without using strcat function.
    c) Concatenate 2 string using strcat function.
    d) Compare 2 strings
    e) Calculate length of the string (use pointers)
    f) To convert all lowercase characters to uppercase
    g) To convert all uppercase characters to lowercase
    h) Calculate number of vowels
    i) To reverse the string
11. Create a class called Rational for performing arithmetic with rational nos. Use integer variables to represent numerator and denominator. Write a constructer method that enables an object of this class to be initialized when it is declared. Provide a no argument constructer with default values in case no initializes are provided. Provide a copy constructor also. Write methods for

a. Addition, Overload ‘+’ operator
b. Subtraction, Overload ‘-’ operator
c. Multiply, Overload ‘*’ operator
d. Division, Overload ‘/’ operator
e. Increment, Overload ++ operator (both prefix & postfix)
f. Decrement, Overload -- operator (both prefix & postfix)
g. Overload operator == (to check equality of two rational nos.) as a friend function
h. Overload Assignment operator
i. Printing in the form of a/b

WAP to test the above class.

12. Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.
13. Write a macro to test whether a year is leap or not.
14. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create Employee class. Maintain Employee information, i.e. create, display and delete employee objects and calculate their wages. (Show Runtime Polymorphism).
15. 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)
    f) Overload Assignment operator
    g) Check if it is a Cube or cuboid

    WAP to test the above class.
16. 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 +).
17. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
18. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
19. Perform Stack operations using Linked List implementation.
20. Perform Stack operations using Array implementation. Use Templates.
21. Perform Queues operations using Circular Array implementation. Use Templates.
22. Create and perform different operations on Double-ended Queues using Linked List implementation.
23. WAP to scan a polynomial using linked list and add two polynomial.
24. WAP to calculate factorial and to compute the factors of a given no. (i)using recursion, (ii) using iteration
25. (ii) WAP to display fibonacci series (i)using recursion, (ii) using iteration
26. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion
27. WAP to create a Binary Search Tree and include following operations in tree:
    (a) Insertion
    (b) Deletion by copying
    (c) Deletion by Merging
    (d) Search a no. in BST
    (e) Display its preorder, postorder and inorder traversals
    (f) Display its level-by-level traversals

(g) Count the non-leaf nodes and leaf nodes
(h) Display height of tree
(i) Create a mirror image of tree

28. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
29. WAP to reverse the order of the elements in the stack using additional stack.
30. WAP to reverse the order of the elements in the stack using additional Queue.
31. WAP to implement Diagonal Matrix using one-dimensional array.
32. WAP to implement Lower Triangular Matrix using one-dimensional array.
33. WAP to implement Upper Triangular Matrix using one-dimensional array.
34. WAP to implement Symmetric Matrix using one-dimensional array.

207- Digital Electronics practical lab based on 104 of First Semester

Section A:

1. Minimization a Boolean expression with 5-variables and realize the circuit with universal gates.
2. Design a half-adder and full-adder and verify the truth table.
3. Design a half-subtractor and full-subtractor and verify the truth table.
4. Design a 4-bit adder-subtractor circuit.
5. Design a 4-bit word parity generator / checker circuit.
6. Design a Binary to Gray code Converter circuit.
7. Design a BCD-to-Excess-3 code converter circuit.
8. Design a 4-bit digital comparator (Magnitude Comparator) circuit.
9. Design a 4 to 1 line multiplexor and verify the truth table.
10. Design a 2 to 4 line decoder with enable input and verify the truth table. Convert the decoder into 1*4 demultiplexer.
11. Design a clocked MS JK flip-flop and verify the truth table.

Section B:

1. Design a synchronous mod 6 counter and verify the truth table.(Use flip-flop)
2. Design a ripple mod- 10 counter and verify the truth table.(Use flip-flop)
3. Design a serial-in-parallel-out 4-bit shift register and verify the truth table.(Use flip-flop)
4. Design a parallel-in-serial-out 4-bit shift register and verify the truth table.(Use flip-flop)
5. Design a 16X4 RAM circuit.

Note: In the examination two exercises one from each section will be set.