Shri B.V.V. Sangha's

BASAVESHWAR ENGINEERING COLLEGE (AUTONOMOUS), BAGALKOT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Syllabus for M. Tech. (CSE) for the 2017-18 Admitted Batch

Semester-I:

Sl.No	Subject Code	Subjects Name	Hours / Week			Credits	Exam Marks		
			L	T	P		CIE	SEE	Total
1	PCS121C	Advanced Algorithms	4	0	0	4	50	50	100
2	PCS122C	Advances in Operating Systems	4	0	0	4	50	50	100
3	PCS123C	Effective Teaching Learning Practices	2	0	0	2	50	50	100
4	PCS007E	Elective-1 (Distributed Systems)	4	0	0	4	50	50	100
5	PCS034E	Elective-2 (Cloud Computing)	4	0	0	4	50	50	100
6	PCS036E	Elective-3 (Big Data Analytics)	4	0	0	4	50	50	100
7	PCS124S	Seminar	0	2	2	2	50	50	100
8	PCS125L	Computing Lab-1	0	1	3	2	50	50	100
Total			**22**	**02**	**02**	**26**	**400**	**400**	**800**

ADVANCED ALGORITHMS

Sub Code	**:**	**PCS121C**	**Credits**	**:**	**04**
Contact Hours	**:**	**04**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**48**	**SEE Marks**	**:**	**50**

COURSE OBJECTIVES

Pre-requisites: Analysis and Design of algorithms

- Determine the most suitable algorithm for any given task and then apply it to the problem.
- Design and implement efficient solutions to various real world problems through algorithms.
- Develop mathematical skills for algorithm design, analysis, and evaluation
- Explain more complex algorithms and proofs in written form,
- Analyze various algorithms for efficiency.

Course Outcomes:

At the end of the course the student will be

- Having knowledge of the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate, and analyze them.
- Understand the difference between a randomized algorithm and an algorithm with probabilistic inputs and the different ways to analyze randomized algorithms (expected running time, probability of error). Recite algorithms that employ randomization.
- Analyze randomized algorithms. Employ indicator random variables and linearity of expectation to perform the analyses. Recite analyses of algorithms that employ this method of analysis.
- Describe the different methods of amortized analysis (aggregate analysis, accounting and potential method). Perform amortized analysis.
- Explain major string matching algorithms and compare efficiencies of different algorithms. And Gain a good understanding on a wide range of advanced algorithmic problems, their relations and variants, and application to real-world problems.

UNIT-I (12 Hours)

Amortized Analysis: Aggregate, Accounting and Potential Methods. **Graph Algorithms:**

Bellman - Ford Algorithm; Johnson's Algorithm for sparse graphs; Flow networks and Ford-Fulkerson method; Maximum bipartite matching. **Polynomials and the FFT :**Representation of polynomials; the DFT and FFT; Efficient implementation of FFT

UNIT-II (12 Hours)

Number -Theoretic Algorithms: Elementary notions; GCD; Modular Arithmetic; Solving modular linear equations; The Chinese remainder theorem; Powers of an element; RSA cryptosystem; Primality testing; Integer factorization. **String-Matching Algorithms:** Naïve string Matching; Rabin - Karp algorithm; String matching with finite automata; Knuth-Morris-Pratt algorithm Boyer – Moore algorithms.

UNIT-III (12 Hours)

Probabilistic and Randomized Algorithms: Probabilistic Algorithms, randomizing deterministic algorithms, Monte Carlo and Las Vegas Algorithms, Probabilistic numerical algorithms, Probabilistic parallel algorithms. NP-Complete Problems: The classes P and NP, Reducibility**,** NP- complete problems: Cook's theorem, Sample NP-complete problems, the class co-NP, The Classes NC and P-Complete.**Approximation Algorithms:** Bin Packing, The Steiner tree problem, the facility location problem.

UNIT-IV (12 Hours)

Introduction to parallel algorithms and architectures: Approaches to the design of parallel algorithms, Architectural constraints and design of parallel algorithms, Performance measures of parallel algorithms, parallel sorting. **Internet algorithms:** Search Engines, Ranking web pages, Hashing, Caching, content delivery and consistent hashing, Message security algorithms.

Note: A list of assignments will be provided in the beginning of semester and evaluated for 20 Marks

TEXT BOOKS:

1. T. H Cormen, C E Leiserson, R L Rivest and C Stein: **"Introduction to Algorithms"**, 2nd Edition, Prentice-Hall of India, 2002.
2. Kenneth A. Berman and Jerome L. Paul: **"Algorithms"**, Cengage Learning, 2002.

REFERENCE BOOKS:

1. Ellis Horowitz, Sartaj Sahni, S.Rajasekharan: " **Fundamentals of Computer Algorithms"** , 2nd Edition, University Press, 2007.
2. Alfred V. Aho,John E. Hopcroft, J.D.Ullman: " **The Design and Analysis of Computer Algorithms"** , Addison-Wesley, 1974.

ADVANCES IN OPERATING SYSTEMS

Sub Code	**:**	**PCS122C**	**Credits**	**:**	**04**
Contact Hours	**:**	**04**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**48**	**SEE Marks**	**:**	**50**

COURSE OBJECTIVES

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication
3. To learn the mechanisms involved in memory management in contemporary OS
4. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
5. To know the components and management aspects of concurrency management

COURSE OUTCOMES

At the end of the course the student will be
1. Exposed to different operating systems and advancements
2. Familiar with the concepts like multithreading and synchronization.
3. Having the knowledge of scheduling in uniprocessor and multiprocessor systems.
4. Realizing the importance of security in operating systems.
5. Having the knowledge of distributed, client-server and cluster environments.

UNIT-I (12 Hours)

Operating System Overview; Operating System Objectives and Functions, The Evolution of Operating Systems, Major Achievements, Developments Leading to Modern Operating Systems, Microsoft Windows Overview, Traditional UNIX Systems, Modern UNIX Systems, Linux. **Process Description and Control;** What is a Process? Process States, Process Description, Execution of the Operating System, Security Issues, UNIX SVR4 Process Management, **Threads, SMP, and Microkernels;** Processes and Threads, Symmetric Multiprocessing (SMP), Microkernels, Windows Vista Thread and SMP Management, Linux Process and Thread Managements.

UNIT-II (12 Hours)

Concurrency: Mutual Exclusion and Synchronization; Principles of Concurrency, Mutual Execution: Hardware Support, Samaphores, Monitors, Message Passing, Readers/Writers Problem, **Concurrency: Deadlock and Starvation;** Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, An Integrated Deadlock Strategy, Dining Philosophers Problems, Dining Philosophers Problems, Linux Kernel Concurrency Mechanisms, Windows Vista Concurrency Mechanisms

UNIT-III (12 Hours)

Uniprocessor Scheduling; Types of Scheduling, Scheduling Algorithms, Traditional UNIX Scheduling, **Multiprocessor and Real-Time Scheduling;** Multiprocessor Scheduling, Real-Time Scheduling, Linux Scheduling, UNIX FreeBSD Scheduling, Windows Vista Scheduling, **Embedded Operating Systems;** Embedded Systems, Characteristics of Embedded Operating Systems, eCOS, TinyOS.

UNIT-IV (12 Hours)

Computer Security Threats; Computer security concepts, Threats, Attacks, and Assets, Intruders, Malicious software overview, Viruses, worms, and bots, Rootkits, **Computer Security Techniques;** Authentication, Access Control, Intrusion Detection, Malware Defense, Dealing with Buffer Overflow Attacks, Windows Vista Security. **Distributed Processing, Client/server and Clusters;** Client/server Computing, Distributed Message Passing, Remote Procedure, Clusters, Windows Vista Clusters Server, Sun Cluster.

TEXT BOOK:

1. William Stallings, "Operating Systems: Internals Design and Principles", 6th edition, Longman, 2009.

REFERENCES:

1. Gary Nut, "Operating Systems", Third Edition, Pearson Education. 2006.
2. Mukesh Singhal, Niranjan Shivaratri, "Advanced Concepts in Operating Systems – Distributed, Database and Multiprocessor Operating Systems", Tata McGraw-Hill, 2001.
3. Rajib Mall, "Real-Time Systems: Theory and Practice", Prentice Hall, 2006.
4. Andrew S. Tanenbaum and Herbert Bos, "Modern Operating Systems", Fourth Edition, Prentice Hall, 2014.
5. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", Wiley, Eighth edition, 2008.

EFFECTIVE TEACHING LEARNING PRACTICES

Sub Code	**:**	**PCS123C**	**Credits**	**:**	**02**
Hours/Week	**:**	**02**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**24**	**SEE Marks**	**:**	**50**

COURSE OUTCOMES

- Design Instruction to teaching practice.
- Create effective Learning environment.
- Implement Instructions to teaching practice.
- Assesses and Communicates Learning Results.
- Asses Teaching/Learning skills.

Unit-I (6 Hours)

Current Developments in Education: introduction, Standards and targets, Stress in teaching, Indiscipline and bullying, School diversity, changing the nature of teaching.
Aims, Objectives and Intended Learning Outcomes: Two kinds of objectives and intended learning outcomes: Behavioral objectives, Non-behavioral objectives, objectives and intended learning outcomes in individual learning.

Unit-II (6 Hours)

Learning and Teaching: Constructivist theory, Higher-order thinking, The brain and learning, Neuroscience and working memory, Metacognition, Learning styles, Motivation, Co-operative learning, key characteristics of effective teaching, Key questions for teaching skills, Non-verbal teacher communication, Modeling, Student teachers attitudes and expectations and the influence they exert on classroom behavior, The organization of learning.

Unit-III (6 Hours)

Assessment: The context of assessment, Assessment in the early years, Assessing pupils' progress, Assessment for learning, The purposes of assessment, The types of assessment, Reliability and validity in assessments, Methods of gathering assessment data, Written sources of data collection, Non-written sources of data collection, Providing opportunities for assessment, Designing an assessment task, Marking work, a worked example of an assessment activity

Unit-IV (6 Hours)

Record Keeping and Report Writing: The purposes of record keeping, the use of the record for reporting purposes, The formality of the record, The contents of the record, the audiences of the record, the style and format of the record, the timing of the record entry, writing reports, ABET Standard: For engineering teaching

Text Book:

1. "A Guide To Teaching Practice", Louis Cohen, Lawrence Manion, Keith Morrison and Dominic Wyse, 5th Edition, Companion@ Website.

REFERENCE BOOKS:

1. **A Taxonomy for Learning Teaching and Assessing,** Lorin W. Anderson and David R. Krathwohl, Pearson Education
2. Strategies for Effective Teaching, Allan C. Ornstein, McGrawHill

Theory	**-**	**2 Hours**
Practical (Delivery + Material preparation)	**-**	**2 Hours**

Allotment of topics/subject/delivery sessions:

- A student is associated with a subject teacher.
- The subject teacher will assign 8 sessions (1 per week)
- The teacher will assess the teaching material (including handouts) prior to the delivery.
- The teacher will evaluate his delivery for every session and keep records.
- The teacher will submit internal evaluation report at the end of semester

Conduction of CIE:

- Two CIE's based on theory (30 Marks)
- Internal evaluation of teachers (20 Marks 15 CIE Presentation + 5 Marks assignments)

Conduction of SEE:

- Theory (50% of Marks) Examination for 100 Marks to be reduced to 50 Marks.

DISTRIBUTED SYSTEMS

Sub Code	**:**	**PCS007E**	**Credits**	**:**	**04**
Hours/Week	**:**	**04**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**48**	**SEE Marks**	**:**	**50**

COURSE OUTCOMES

At the end of the course the student will be able to

- Define distributed system, list its characteristics, examples, challenges and address the issues involved in designing the distributed system.
- Describe the successful approaches in the form of abstract models, algorithms and do the detailed case studies of widely-used systems.
- Understand the various processes such as interprocess communication, Remote procedure call, and Remote method invocation and related issues.
- List out the additional needs of operating system to support distributed systems and explain various mechanisms to provide security.
- Understand the various distributed systems such as distributed file systems, distributed transactions, distributed shared memory and analyze the associated case studies.

UNIT – I (12 Hours)

Characterization of Distributed Systems and System Models: Introduction, Examples of distributed systems, Resource sharing and the Web, Challenges, Architectural models, Fundamental models. Networking and Internetworking**:** Types of Networks, Networks principles, Internet protocols, Network case Studies (Ethernet, wireless LAN and ATM). Interprocess Communication: Introduction, The API for the Internet protocols, External data representation and marshalling, Client-Server communication, Group communication

UNIT – II (12 Hours)

Distributed Objects and Remote Invocation: Communication between distributed objects, Remote procedure call, events and notifications, JAVA RMI case study, Operating System Support and Security**:** The Operating system layer, protection, processes and threads, communication and invocation, operating system architecture, overview of security techniques, cryptographic algorithms, digital signatures, cryptography pragmatics, case studies: Needham-Schroeder, Kerberos, SSL and Millicent.

UNIT – III (12 Hours)

Distributed File Systems: File service architecture, Sun Network file system, Andrew file system, Recent advances. Transactions and Concurrency Control**:** Transactions, nested transactions, locks, optimistic concurrency control, timestamp ordering, comparison of methods for concurrency control.

UNIT – IV (12 Hours)

Distributed Transactions**:** Flat and nested distributed transactions, atomic commit protocols, concurrency control in distributed transactions, distributed deadlocks, transaction recovery. Distributed Shared Memory**:** Design and Implementation issues, sequential consistency and Ivy, Release consistency and Munin, other consistency models, CASE Studies: COBRA, Mach

TEXT BOOK:

1. George Coulouris, Jean Dollimore, Tim Kindberg: “Distributed Systems, Concept and Design”, 3rd edition, Pearson Education, 2005.

REFERENCE BOOK:

1. Andrew S. Tanenbaum & Marten van Steen, Distributed Systems – Principles and Paradigms, PHI, 2002.

CLOUD COMPUTING

Sub Code	**:**	**PCS034E**	**Credits**	**:**	**04**
Hours/Week	**:**	**04**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**48**	**SEE Marks**	**:**	**50**

COURSE OUTCOMES

At the end of the course the student will be able to

- Understand the significance, characteristics and challenges of cloud computing.
- Comprehend the reasons behind the popularity of Cloud Computing Technology, accessing the Clouds and Developing Applications
- Explore the Local Clouds, thin Clients, best practices, future and evolution of Cloud computing,
- Comprehend different cloud computing platforms and their architectural features.
- Explain cloud services and its usage for individuals, differentiate between cloud services and cloud platforms and its features.

Unit – I (12 hours)

Cloud Computing Basics: Overview, Applications, Intranet and the Cloud, First Movers in the Cloud; The Use of Cloud Computing, Benefits, Security concerns, regulatory issues; Overview of different cloud computing applications that are implemented; Business case for implementing a Cloud: Cloud Computing Services, Applications help to the business, deleting the datacenter, Saleforce.com, Thomson Reuters.

Unit – II (12 hours)

Cloud Computing Technology: Hardware and Infrastructure: Clients, Security, Network, Services; Accessing the Clouds: Platforms, Web applications, Web APIs, Web Browsers; Cloud Storage: Overview, Cloud Storage providers, Standards: Applications, Client, Infrastructure, Service.

Unit – III (12 hours)

Cloud Computing at Work: Software as a service: Overview, Driving Forces, Company offerings, Industries; Software plus services: Overview, Mobile Device Integration, Providers, Microsoft Online; Developing Applications: Google, Microsoft, Intuit QuickBase, Cast Iron Cloud, Bungee Connect, Development: Google, Sales Force, Azure.

Unit – IV (12 hours)

Local Clouds and Thin Clients: Virtualization, server solutions, Thin Clients; Migrating to the clouds: Cloud services for individuals, Cloud services aimed at Mid-market, and Enterprise-Class, Migration; Best practices and the future of Cloud computing: analyzing the services, Best practices, How Cloud Computing might evolve.

TEXT BOOKS:

1. Cloud Computing a Practical approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, Tata McGraw-HILL, 2010 Edition

REFERENCE BOOKS:

1. Mastering Cloud Computing, Rajkumar Buyya, Christian Vecchiola, S. Thamari Selvi, McGraw Hill Education (India) Private Limited.
2. Distributed and Cloud Computing, Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra, Morgan Kaufmann Publishers 2012.
3. Cloud computing, Barrie Sosinsky, Wiley India.
4. Cloud Computing, Kumar Saurabh, 2nd Edition, Wiley, India

BIG DATA AND ANALYTICS

Sub Code	**:**	**PCS036E**	**Credits**	**:**	**04**
Hours/Week	**:**	**04**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**48**	**SEE Marks**	**:**	**50**

COURSE OBJECTIVES

- Understand the need to integrate structured, semi-structured and unstructured data.
- Analyzing the reasons activities of Hadoop Ecosystem and its versions.
- Comprehend the significance of NoSQL and NewSQL databases.
- Able to perform MapReduce and HDFS operations.
- Familiar with MongoDB, Cassandra, Hive and Pig frameworks.

COURSE OUTCOMES

Pre-requisites: Basic knowledge of Relational Database Management System concepts

- Understand the significance, characteristics and challenges of big data.
- Comprehend the reasons behind the popularity of Hadoop Ecosystem and its versions.
- Identify the significance of NoSQL and NewSQL databases.
- Understand the application of MongoDB and Cassandra frameworks.
- Comprehend the Hive and Pig frameworks

UNIT- I (12 HOURS)

(Types of digital data): Types of Digital Data, Structured: Sources of structured data, Ease with Structured data, Semi-Structured: Sources of semi-structured data, Unstructured: Sources of unstructured data, Issues with terminology, Dealing with unstructured data,**(Big Data):** Characteristics of data, What big data? Definitions and Challenges of big data, other characteristics of data which are not definitional traits of big data, Why big data? Are we just an information consumer or do we also produce information? Traditional Business Intelligence(BI) versus Big data, A typical BI environment, A big data environment, Big data stack, What is changing in the realms of big data?

(Big data analytics)

Where do we begin? What is big data analytics? What big data analytics isn't? Why this sudden hype around big data analytics? Classification of analytics Top challenges facing big data, Why is big data analytics important? Greatest challenges that prevent businesses from capitalizing on big data, what kind of technologies are we looking towards to help meet the challenges posed by big data? Data science, Data Scientist – your new best friend!!!, Terminologies used in big data environment, In memory analytics, In database processing, Massively parallel processing, Parallel versus distributed systems, Shared Memory architecture, Consistency, Availability, Partition Tolerance (CAP) theorem explained, Basically Available Soft State Eventual Consistency (BASE), Few top Analytics tools, Introduction to Jasper Report using Jasper Soft Studio

UNIT- II (12 HOURS)

(The big data technology landscape): NoSQL, Where is it used? What is it? Types of NoSQL databases, Why NoSQL? Advantages of NoSQL, What we miss with NoSQL? NoSQL Vendors, SQL Versus NoSQL , NewSQL, Comparison of SQL, NoSQL and NewSQL, Hadoop: Features of Hadoop, Key advantages of Hadoop,Versions of Hadoop, Hadoop 1.0, Hadoop 2.0, Overview of Hadoop Ecosystems, Hadoop Versus, SQL, Integrated Hadoop systems offered by leading market vendors,

Cloud based Hadoop solutions. **(Hadoop) :** Introducing Hadoop, Why not RDBMS, Distributed Computing Challenges, Brief History of Hadoop, Hadoop Overview, Hadoop Components, High Level Architecture of Hadoop, Hadoop Distributed File System(HDFS), HDFS Architecture, Daemons Related to HDFS, Working with HDFS Command, Special Features of Hadoop, Processing Data With Hadoop,Introduction, How Map Reduce Works? Map Reduce Example, Word Count Example using Java,

Managing Resources and Applications with YARN, Introduction, Limitation of Hadoop 1.0, Hadoop 2: HDFS, Hadoop 2: YARN, Interacting with Hadoop EcoSystem, Hive,Pig, HBASE, Sqoop, Business Intelligence on Hadoop.

UNIT- III (12 HOURS)

(NoSQL - MongoDB): What is MongoDB? Why MongoDB? Using JSON, Creating or generating a unique key, Support for dynamic queries, Storing binary data, Replication, Sharding, Updating information in-place, Terms used in RDBMS and MongoDB, Data types in MongoDB, MongoDB - CRUD (Insert(), Update(), Save(), Remove(), find()), MongoDB- Arrays, Java Scripts, Cursors, Map Reduce Programming, Aggregations.

(NoSQL - Cassandra): What is Cassandra? Why Cassandra? Peer to peer network, Gossip and Failure detection, Anti-Entropy & Read Repair, Writes in Cassandra, Hinted handoffs, Tunable consistency,Cassandra- CQLSH - CRUD, Counter, List, Set, Map, Tracing.

UNIT- IV (12 HOURS)

(Hadoop Hive): Introduction to Hive - The Problem, Solution - Hive Use Case, Data Growth, Schema Flexibility and Evolution, Extensibility, What is Hive, History of Hive and Recent Releases of Hive, Hive Features, Hive Integration and Work Flow, Hive Data Units, Hive Architecture, Hive Primitive Data Types and Collection Types, Hive File Formats, Hive Query Language - Statements, DDL , DML, Hive Partitions, Bucketing, Views, Sub Query, Joins, Hive User Defined Function, Aggregations in Hive, Group by and Having, Serialization and Deserialization, Hive Analytic Functions.

(Hadoop - Pig): Introducing Pig, History and Anatomy of Pig, Pig on Hadoop, Pig Features, Pig Philosophy, Word count example using Pig, Use Case for Pig, Pig Primitive Data Types, Colletion Types and NULL, Pig Latin Overview, Pig Latin Grammar - Comments, Keywords, Identifiers, Case sensitivity in Pig, Common Operators in Pig, Pig Statements, LOAD, STORE, DUMP, Interactive Shell - GRUNT, FILTER,SORT,GROUP BY,ORDER BY, JOIN, LIMIT, Pig Latin Script, Local Mode,Map Reduce Mode, Running Pig Script,Working with Field, Tuple, Bag,User Defined Function, Parameters in Pig.

TEXT BOOK:

1. Big Data and Analytics, Seema Acharya and Subhashini Chellappan – Wiley India, 2015.

REFERENCE BOOKS:

1. Frank J Ohlhorst, “Big Data Analytics: Turning Big Data into Big Money”, Wiley and SAS Business Series, 2012.
2. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
3. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch , James Giles, David Corrigan, “Harness the Power of Big data – The big data platform”, McGraw Hill, 2012.

SEMINAR

Sub Code	**:**	**PCS124S**	**Credits**	**:**	**02**
Hours/Week	**:**	**-**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**-**	**SEE Marks**	**:**	**50**

COURSE OUTCOMES

- Access information in a variety of ways appropriate to a discipline, including locating and using library collections and services and other search tools and databases.
- Obtain, select, store, create, use and organize support materials appropriately.
- Demonstrate effective writing skills and processes by employing the rhetorical techniques of academic writing, including invention, research and critical analysis.
- Identify and critically evaluate the quality of claims, explanation, support, and document it in professional way.

A student doing research under supervision of a professor for his/her project work discusses his/her survey, findings, conclusions and new technologies with professor and presents it to complete class. The learning's of seminar course introduced in our curriculum meets the objectives of programme (i,iii,v,ix,x,xi) and bring in collection, distributing, analyzing and communication capability into learner.

COMPUTING LAB-1

Sub Code	**:**	**PCS125L**	**Credits**	**:**	**02**
Contact Hours	**:**	**-**	**CIE Marks**	**:**	**50**
Total Hours	**:**	**-**	**SEE Marks**	**:**	**50**

LAB ASSIGNMENT LIST FOR ADVANCED ALGORITHMS

Note: The following programs can be executed on any tool / language.

1. Design and write a program to implement Extended Euclid's algorithm to compute the, greatest common divisor of integers *a* and *b*, also the coefficients of Bézout's identity, which are integers *x* and *y* such that ax+ by= gcd (a,b).
2. Design and write a program to implement a Miller Rabin / Monte Carlo algorithm to test the primality of a given integer and determine its performance.
3. Design and write a program to calculate pow(x,n) i.e for given two integers x and n, compute x^n. Assume that x and n are small and overflow doesn't happen.
4. Design and write a program to implement the Bellman-Ford algorithm to solve the single-source shortest-paths problem and determine its performance.
5. Design and write a program to implement Johnson's algorithm to solve the all pairs shortest path problem, i.e. given an input graph with general edge weights (can be negative) with no negative cycles, find the shortest (u, w) path for all pairs of vertices (u, w). If the input graph has any negative cycles, the program will report this.
6. Design and write a program to implement Ford Fulkerson algorithm to find maximum flow and determine its performance.
7. Design and write a program to solve string matching problem using naïve approach and Boyer Moore approach. Compare the performance.
8. Design and write a program to solve string matching problem using the KMP algorithm. Determine the performance.
9. Design and write program to solve string matching problem using Robin Karp algorithm and determine its performance.
10. Design and write a program to solve string matching problem using Finite Automata and determine its performance.