I'm not robot
reCAPTCHA
Continue

Cell division lesson plan middle school

According to the new curriculum of Mumbai University, forS.Y.B.Sc. (Computer Science), IV. (Computer Science) Assistant Professor, Department of Computer Science,N.G. Acharya & D.K. Marathe College,Chembur, Mumbai.ISO 9001:2008 CERTIFIEDFundamentals ofAlgorithms ©AuthorsNot part of the publication may not be reproduced, stored in a search engine, or transmitted in any form or in any way, electronic, mechanical, photocopying, recording and/or otherwise without prior written permission from the publisher. First edition : 2018Appeared by Mrs. Meena Pandey of Himalayan Publisher Pvt. Ltd.,Ramdoot, Dr. Bhalerao Marg, Girgaon, Mumbai - 400 004.Phone: 022-23860170, 23863863; Fax: 022-23877178E-mail: himpub@vsnl.com; Website: www.himpub.comBranch Offices : New Delhi : Pooja Apartments, 4-B, Murari Lal Street, Ansari Road, Darya Ganj, New Delhi - 110 002. Telephone: 011-23270392, 23278631; Fax: 011-23256286Nagpur : Kundanlal Chandak Industrial Estate, Ghat Road, Nagpur - 440 018.Phone: 0712-2738731, 3296733; Fax: 0712-2721216Bengaluru : Plot No. 91-33, 2nd Main Road, Seshadripuram behind Nataraja Theatre, Bengaluru - 560 020. Phone Number: 080-41138821; Mobile: 09379847017, 09379847005Hyderabad : No. 3-4-184, Lingampally Next to Raghavendra Swamy Matham, Kachiguda,Hyderabad - 500 027. Phone: 040-27560041, 27550139Chennai : New No. 48/2, Old No. 28/2, Ground Floor, Sarangapani Street, T. Nagar,Chennai - 600 012. Mobile: 09380460419Pune : Laksha Apartment, First Floor, No. 527, Mehunpura, Shaniwarpeth (NearPrabhat Theatre), Pune - 411 030. Telephone: 020-24496323, 24496333; Mobile: 09370579333Lucknow : House No. 731, Shekhupura Colony, Near BD Monastery School, Aliganj, Lucknow - 226 022. Phone Number: 0522-4012353; Mobile: 09307501549Ahmed aba d : 114, SHAIL, 1st floor, Opp. Madhu Sudan House, C.G. Road, NavrangPura, Ahmedabad - 380 009. Phone Number: 079-26560126; Mobile: 09377088847Ernakulam : 39/176 (New No. 60/251) 1st Floor, Karikkamuri Road, Ernakulam,Kochi - 682 011. Phone: 0484-2378012, 2378016; Mobile: 09387122121Bhubaneswar : Plot No. 214/1342, Budheswari Colony behind Durga Mandap,Bhubaneswar - 751 006. Phone number: 0674-2575129; Mobile: 09338746007Kolkata : 108/4, Beliaghata Main Road, NEAR ID Hospital, Opp. SBI Bank, Kolkata - 700 010. Phone Number: 033-32449649; Mobile: 07439040301DTP by Sagar YadavPrinted a : Geetanjali Press Pvt. Ltd., Nagpur. On behalf of hph. I dedicate this book to my dearest daughter, Anushka, and her son Atharva for their unconditional love, which has made me stronger, better and more complete than I ever imagined. Archana JadhavI would like to dedicate this book to my father Vishwas Kashid, who motivated me in his words: The result of success daily efforts. I also dedicate this book to Mother Rukhamini Kashid, whose loving soul still sustains me. Last but not least, I express my gratitude to my dearest brother and sister for their love and support. I would also like to thank my co-author, Archana Jadhav, for believing in me and giving me this golden opportunity. Sheetal Kashid We would like to express our sincere thanks to Mr. S.K. Srivastava of Himalayan Publishing House Pvt. Ltd., giving us the opportunity to write this book. Foreword We are sure that this book will satisfy all your needs on the subject. As a step towards improving and improving the book, suggestions from teachers and students are welcome. Please send letters to archana.sjd@gmail.com and kashid.sheetal3@gmail.comFinally, we would like to acknowledge our sincere respect and gratitude to Kiran Gurbani (Head of Computer Science and IT Department, RK Talreja College, Ulhasnagar) reviewing this book thoroughly and providing an environment that encourages new thinking and innovations and his support that has helped us to get out of this book in time. We are grateful to Shri S.K. Srivastava and the Himalayan Publishing House for their cooperation. AuthorsIt gives us great pleasure to submit a book on the basics of algoritmusokforS.Y.B.Sc. (Sem.IV) ComputerScience.Thisbookisbasedonnewsyllabus prescribedMumba University 2017-18. The main topics have been divided into subtopics for easy understanding. Many solved examples have been added to the book by implementing them in Python. Mcq's is provided to improve aptitude skills. Syllabus Goals:1. To understand the principles of algorithm design and why algorithm analysis2 is important. To understand how algorithms are implemented in python3. To understand how to transform new problems algorithmic problems with effectivesolutions4. Understand algorithm design techniques to solve different problemsUnit Number of Modules/UnitsUnit 1 Introduction to the algorithm, Why should you analyze the algorithm?, Run time analysis, How to compare algorithms?, Growth rate, Commonly used growth rates, Types of analyses, Asymptotic Marking, Big-O Notation, Omega-)Notation, Theta-şNotation,Asymptotic Analysis, Properties of Notations, Commonly Used Logarithms and Summations, Performance Characteristics Algorithms, Master Theorem's Divide andConquer, Divide and Conquer Master Theorem: Problems and Solutions, MasterTheorem's Withdrawal and Conquer Repetitions, Method of Guessing and Confirming. Unit 2 tree algorithms: What is wood? Vocafe, binary trees, types of binary trees, properties of binary trees, binary tree passages, generic trees (N-ary trees), stitched binary tree passages, term trees, binary search trees (BSTs), balanced binary search trees, AVL (Adelson-Velskii and Landis) Trees.Graph Vocara, Graph Applications, GraphRepresentation, Graph Traversals, Topological Sorting, Shortest Path Algorithms, MinimalSpanning Tree.Selection Algorithms: What are selection algorithms? Selection sorting, Partition-based selection algorithm, linear selection algorithm - median median jats laying algorithms,finding the K smallest elements arranged in order. Unit 3 algorithms design techniques: introduction, classification, classification by Implementation method, classification by Design Method.Greedy Algorithms: Introduction, greedy strategy, elements of greedy algorithms, advantages and disadvantages of the greedy method, greedy applications, UnderstandingGreedy Technique.Fundamentals algorithms Divide and conquer: Introduction, What is the gap and conquest? ,Divide and conquer visualization, understanding Divide and Conquer, Advantages of Divide and Conquer, Disadvantages of Divide and Conquer, Master Theorem, Divideand Conquer Applications.Dynamic Programming: Introduction, What is dynamic programming strategy?, Features of Dynamic Programming Strategy, Problems That Can Be Solved by Dynamic Programming, Dynamic Programming Approaches, Examples of DynamicProgramming Algorithms, Understanding Dynamic Programming, Longest CommonSubsequence.List of Practicals1. Write a Python program for matrix multiplication. Discuss the complexity of the algorithm you are using.2. Write a Python program to sort n names using the Quick sort algorithm. Discuss the complexity of the algorithm you are using.3. Write a Python program to sort n numbers using the Merge sort algorithm. Discuss the complexity of the algorithm you are using.4. Write a Python program to insert an element into a binary tree.5. Write a Python program to delete an item (assuming data is specified) from a binary tree.6. Write a Python program to verify that a given G graph has a simple path from source to destination d. Let's say graph G is displayed using the adjacent matrix.7. You can write a Python program to find the smallest and largest elements of the selection algorithm in the A-size array. Discuss the complexity of time.8. You can write a Python program to find the second largest element of the N-size Array A using the Tournament method. Discuss the complexity of time.9. Write a Python program to implement the Huffman encoding algorithm. Discuss the complexity of the algorithm.10. Write a Python program to multiply Strassen Matrix using the Divide andConquer method. Discuss the complexity of the algorithm. Question paper pattern(Time: 21/2 Hours) [Total marks: 75]N.B. 1. All questions are mandatory.2. The diagrams on the right indicate signs. Q.1 Attempt all (All 5Marks) (15M) (a) Select the correct answer as follows: (5)(b) Fill in the blank: (5) (c) Enter the following: (5)Q.2 Attempt the (Any THREE) Unit I (Any THREE) is unit I (Any THREE) unit I (Any THREE) unit I (15M)(a)(b)(c)(c)(c)(d)(f)(f)Q.3 Attempt following (Any THREE) from Unit II (15M) (a)(b)(c)(d)(d)(f)Q.4 (Any THREE) in Annex III. Name Page No.UNIT I1 Introduction to the algorithm 1 - 122 Asymptotic analysis 13 - 233 Recurrence relationships 24 - 33UNIT II4 Sorting 34 - 505Trees 1 51 - 716 Charts 72 - 108UNIT III7 algorithm design techniques 109 - 1118 Greedy algorithms 112 - 1209 Division and conquer algorithm 121 - 122 610 Dynamic programming 127 - 139Practual 140 - 164 Multiple election questions 165 - 173Ref links 174 - 174 174 Introduction to the algorithm1Structure:1.1 What is algorithm1.2 Algorithm analysis1.3 Comparison of algorithms1.5 Growth rate1.6 Asmptotic notations1.1 What is the algorithm? Algorithmic concept is used in mathematics or computer science or other related subjects. Each algorithm is a list of well-defined statements that perform or perform a specific task. For example: Consider an algorithm that tries to verify whether the battery is working properly or not. Bulb does not workproperlyChargedcellsBattery does not workLoad thecellBuy in newbulbBuy is a newbatteryYESYESNONOFig. 1.1: The battery check flowchart is a graphical representation of an algorithm. A flowchart is used to graphically represent the algorithm. In an algorithm flowchart, you use different types of shapes to implement steps and shapes connected through arrows. The above flowchart shows all possible positions of the battery, whether it works or not. First, check the cells used in the battery, whether it is charged or not. If not, upload the cell. If the cells are charged, make sure the bulb is working properly or not. If it does not work properly, replace the bulb. If the bulb is in a state, the problem is somewhere in the battery, then buy the new bulb. The symbolic representation of FlowchartAmerican National Symbols Institute (ANSI) developed different symbols to graphically represent the algorithm. Table Symbol/Shape Name DescriptionTerminal Table 1.1. 1.1 This is an oval rectangle that represents the beginning of the program. This is a starting and finishing block. Process This is a rectangle that represents a collection of instructions to be executed. Decision It presents it as a diamond. This is a form of decision-making that decides the result on the basis of the condition. Input/Output Indicates a paraleaning gram, which is an input instruction for accepting user input and displaying the result as an outputstatement. Flow Line An arrow indicates the process of instructions.2zThe basics of algorithms 3Intitation into AlgorithmProgram is a set of instructions that the computer performs to perform a specific task. This is the execution of the algorithm. Algorithmic algorithm, which is determined and which ends with the production of the corresponding output with a given input. This is a step-by-step guide to solving a specific task in fini-time. The algorithm is an effective method to solve the problem expressed in the asfinite series of instructions. Each algorithm must meet the following five criteria:1. Input: The algorithm input must be zero or more.2. Output: Using inputs that are highly delivered to an algorithm produces atleast onequantity as output.3. Firmness: The instructions used in the algorithm must be clear and secure, i.e. clear.4. Finality: The algorithm must cancel after a few finitive steps. In all cases, i.e. each algorithm must have a termination or endpoint after the fin final number of steps.5. Efficiency: To complete the task, all instructions must be basic, which can track the instruction using a human being, paper, and pencil method. For example: Write an algorithm and draw a flowchart to add two numbers and store the result.1.2. Figure 2: Flowchart to add two numbersAlgoritmus to add two numbers:Step 1: StartStep 2: Two numbers a and b from the user in step 3. : The StopData structure and the AlgorithmData structure are logical or mathematical models of the specific organization of the data. The mathematical model explains how data is stored in memory and which actions are possible on it. Both the data structure and the algorithm work together. If the set of PivotTable reports does not suggest basic steps to resolve the problem, it is called a procedure. For example, a program always has an algorithm, excepta small difference so that the program does not need to be eliminated. For example: Write an algorithm and draw a flowchart to find the largest of the three numbers. Algorithm:Step 1: StartStep 2: A, b, c.3>. Otherwise, go to step 5.4: If b > c, step 6. Otherwise, go to step 8.5: If you have a > c, goto step7. Otherwise, go to step 8. 6: B is the largest, goto step9. Step 7: The biggest, goto step9. Step 8: c is the largest, goto step9. Step 9: Stop4zFundamentals of Algorithms 5Infusing into the algorithm1.2 ALGORITHM ANALYSIS Analyzing the algorithm means solving an algorithm's problem statement using different approaches. This also means to check the best algorithm. Analysis of a two-phase algorithm:1. Preliminary estimate: Refers to a theoretical mathematical calculation that uses asymptotic markings for representation.2. Back estimate: This is about writing a program, so the operating system, compiler, etc.1.3 OPERATING TIME ANALYSISThe operating time of the algorithm required for specific input operations, based on the number of operations performed. The larger the the longer the number of operations, the longer the the user writes a program, is concerned about the program's resource needs. Algorithm analysis is the definition of time, the amount of memory used, and other resources that the system requires to perform a specific task. The operational time analysis increases the algorithm's execution time (the number of calculation steps) depending on the increase in the input size. Consider the input size indicated as n, and then the time the problem will work is f(n). The run time of an algorithm is measured by the following steps:1. Calculate primary functions: Primary functions are fixed execution time statements. These instructions are as follows: (a) Assign variables to objects(b) Perform arithmetic operations such as adding numbers, subtracting numbers, etc.(c) Comparing two numbers(d) Defining a function and calling the function2. N input size calculation operation: Compare functionf(n)1.4 ALGORITHMS in the algorithm based on time complexity and space complexity. Algorithms are better time complexity and less memory space is a better algorithm. The performance analysis of an algorithm depends on two factors:1. Spatial comm: The amount of memory required to complete the algorithm.2. Time complexity: The time required by the algorithm to complete. The number of steps that the algorithm takes to complete a task calculates the complexity of an algorithm over time. There are three types of steps:(a) Best case: In the best case scenario, calculate the minimum number of steps performed for the specified parameters. (b) Worst case: In the worst case scenario, the maximum number of steps performed is the number of parameters specified. (c) Average case: The average case step number is the average number of parameters specified.1.5 Growth RATE To analyze the algorithm. The higher the amount of data, the higher the amount of resources required by the algorithm. Therefore, a piece of code in the form of the function f(n) shows a rate of resource growth. The growth rates commonly used to analyze the algorithm are: 1. Constant function: A constant resource requirement is one where the resource need does not increase. The simplest function is some fixed c constants, such as c = 10 or c = 2, etc. The constant function is calculated as f(n) = ci.e.; The constant function is used to add two numbers, initializing the value for one variable, comparing two numbers. The graph of such a growth rate is shown by a horizontal line.2. Logarithmic function: A logarithmic growth rate is a growth rate at which resource needs increase by one unit each time the data doubles. The logarithmic function is calculated as follows: f(n) = logœn for b > 1 iff bx = n, where b is called the basis for logarithm.3. Linear function: Another an important function is a linear function that is calculated as f(n) = nN with an input value, and the linear function f assigns n itself. A linear growth rate is a growth rate at which a resource needs data and data that is directly proportional to each other. The growth rate of the linear function is represented by a horizontal line. The example involves comparing a number to each element size nrequire n comparisons.6zFundamentals of Algorithms 7It's introduction to algorithm4. Function N-Log-N: This type of function is calculated as f(n) = nlognThe function assigned to input n is n times the base-2 value of the n logarithm. The log linear growth rate is slightly curved, i.e. lower values than higher ones. The example includes a possible algorithm for sorting n numbers.5. Second-degree function: The rectangular function is calculated as f(n) = n^2The n input value, f is the product of n. The example includes an algorithm with nested loops, where the number of linear instructions for internal loopimplements and outerloop are implemented in a linear number of times. Vertical (value) axis main gridlines6. Cubic function: The cubic function is calculated as f(n) = n^3 with three loops.7. Exponential function:resources. The calculation is made as follows: f(n) = b^nahol b is a constant with a base name, and nis, which is called exponent.1.6 ASYMPTOTIC NOTATIONSAsymptotic, are restrictive behaviors related to the increase in the algorithm's runtime, the size of which is the input limit. There are three types of asymptotic notation:1. Big-O notation: Make it a positive integer f(n) and g(n). This mark is displayed as follows: f(n) = O(g(n)iff has two positive constants c and n0, so f(n) ´cg(n), i.e. f(n) is asymptotically free or equal to g(n) for all values n `n0, where Big-O represents the upper limit. Chart Area 1.` ´3. Therefore, prove that f(n) = O(n) and identify the type of function. Evidence: Place n = g(n)Since it is included in the Big-O notation formula, f(n) ´cg(n)3n + 3 ´c(n) every n `n0. Letn=1,c=4 (3(1)+3)´4(1)6´4, so this condition fails n= 1.Letn=2,c=4 (3(2)+3)´4(2)9´8, this condition again fails n = 2Letn=3,c=4 (3(3)+3)´4(3)12 ´12, so this condition is transferred to n=3Letn=4,c=4 (3(4)+3)´4(4) 1516´ , so this condition is converted to n= 4Now, we can say that 3n + 3 ´cn all values n `3andc=4.Hence, function f(n) = O (n) n `3 is a linear function.1.6.2 Let the function f(n) = 10n² + 5n + 2. Therefore, prove that f(n) = O(n²) and identify the type of function. Evidence: Place n² = g(n)Since it is included in the Big-O notation formula, f(n) ´c. g(n)10n² + 5n + 2 ´c(n²) all n `n0Let n = 1, c = 100 (10(1) + 5(1) + 2) ´100(1)8z Basics of algorithms 9Inst ´100, i.e. this condition is n = 1 ´´. so this condition is passed n= 2.Let n = 3, c = 100 (10(3²) + 5(3) + 2) ´100(3²)107 ´900, so this condition is transferred to n= 3.Now, we can say that 10n² + 5n + 2 ´c² all values n `1 and c = 100.Therefore, this condition is transferred to n = 3.Now , we can say that 10n² + 5n + 2 ´c² all values n `1 and c = 100.Therefore, this condition is transferred to n = 3.Now, we can say that 10n² + 5n + 2 ´c² all values n `1 and c = 100.Therefore, this condition is transferred to n = 3.Now , we can say that 10n² + 5n + 2 ´c² all values n `1 and c = 100.Therefore, this condition is transferred to n = 3.Now , we can say that 10n² + 5n + 2 ´c² all values n `1 and c = 100.Hence this condition is transferred to n = 3.Now, we can say that 10n² + 5n + 2 ´c² , n `1 to f(n²) function is a second-degree function.1.6.3 Let f(n) = 5logn + 6 function. So prove that f(n) = O(log n) and identify the type of function. Evidence: Item log n = g(n)Since it is included in the Big-O marking formula, f(n) ´cg(n)5logn+6´c(log n) all n `n0Let n = 1, c = 100 5 log 1 + 6 ´100.log 15(0) + 6 ´100(0)6´0, so this condition does not meet n = 1.Let n = 2, c = 100 5 log 2 + 6 ´100.log 25(0,30) + 6 ´100(0,30)7,5 ´30 , so this condition does not meet n = 2.Now, we can say that 5 log n + 6 ´c.log n all values n `2 and c = 100 since, n = 1 logn=0.Hence, function f(n) = O(log n) n `2 is a logarithmic function.2. Omega mark: Make it a positive integer f(n) and g(n). This mark is displayed as f(n) = ð(g(n))iff, there are two positive constants c, so that the value f(n) `cg(n), i.e., f(n) ag(n) is greater than or equal to ag(n) at all values n `n0, where ðrepresents the lower limit is 1.` ` ðg 4. Therefore, prove that f(n) = ð(n) and identify the type of function. Evidence: Suppose n = g(n)Since omega is included in the marking formula, f(n) `cg(n)3 + 3 `c(n) all n `n0Letn=1,c=1 (3(1)+3)`1(1)6`1, so this condition is n = 1.Letn=2,c=1 (3(2)+3)`1(2)9`2, so this condition is n = 2.Letn=3,c=1 (3(3)+3)`1(3)12 `3 , so this condition is converted to n= 3.Now, we can say that 3n + 3 `cn all values n `1andc=1.Therefore, function f(n) = ð (n) n`1 is a linear function.1.6.5 Let the function f(n) = n³ + 10 n² + 6. So prove that f(n) = ð(n³) and identify the type of function. Evidence: Place n³ = g(n)Since omega is included in the notation formula, f(n) `cg(n)n3 + 10n² + 6 `c(n3) every n `n0Let n = 1, c = 1 (1³ + 10(1²) + 6) `1(1³)17 `1, so this condition is n = 1.Let n = 2, c = 1 (2³ + 10(22) + 6) `1(2³)54 `8, so this condition is passed n = 2.Let n = 3 , c = 1 (3³ + 10(3²) + 6) `1(3³)123 `27, so this condition is n = 3 ` ð ` `. Theta notation: Make it a positive integer f(n) and g(n). This notation is displayed as follows: f(n)=ş(g(n)f(n)f(n) a large theta g(n.10 zFundamentals of Algorithms 11 Introduction to AlgorithmIff f(n) large O g(n) and f(n) large Omega g(n), i.e., c'g(n) ´f´n] ´cg(n), i.e. isasymptotically equal to g(n) for all n `n00.1.` ` ðg 5. So prove that f(n) = ş(n) and identify the type of function. Evidence: Let g(n) = nf(n) = ş(g(n)), i.e. f(n) be the g(n) Great Theta, if f(n) is the large O,..... and f(n) g(n).c g(n) is a large omega ´f. n) ´cg(n) is n `n0Let n = 1, c' = 1, c' = 1001´3(1) + 3 ´100(1)1´6´100, so this condition is n = 1.Let n = 2, c' = 1, c'' = 100 1(2) ´3(2) + 3 ´100(2)2´9´200, so this condition is n = 2.As in 1.6.1. we have defined that f(n) = O(n) and f(n) = ð(n),i.e.,3n+3=O(n)and+3=ð(n)Therefore, f(n) = ş(n) for n `1 which is the exact limit.1.6.7 Let the f(n) function = 10 n² + 2n + 6. So prove that f(n) = ş(n²) and identify the type of function. Evidence: Let g(n) = n²f(n) = ş(g(n)), i.e. f (n) be the big thete of g(n) if f f n) is the big Oo,..... c'g(n) ´f(n)´cg(n) for all values of n `n0Let n = 1, c' = 1, c'' = 100 1(1²) ´10(1²) + 2(1) + 6 ´100 (1²)1´18 ´100 , i.e. this condition is n = 1.Let n = 2, c' = 1, c'' = 100 1(2²) ´10(2²) + 2(2) + 6 ´100 (2²) 4´50 ´400, so this condition is passed to n = 2 ð ð. , f(n) = ş(n) for n `1, which is the exact boundary. PRACTICE OF QUESTIONS1. Make f(n) = 8n + 12 functions. So, prove that this function identifies f(n) = O(n) and the function type?2. What's an algorithm? How to analyze the algorithm?3. Make it f(n) = 2n³ + 3n² + n + 10 functions. Therefore, prove that this function is f(n) = O(n³) and also the function type.4. Write a comment with asymptotic notation.5. Make f(n) = 8n + 12 functions. Therefore, prove that this function is f(n) = ð(n) and identify the function type.6. Explain the growth rate for the algorithm.7. Make it f(n) = 9n² + 3n + 12 functions. So let's prove that this function is f(n) = ð(n²) and the function type. How do we calculate the run time of an algorithm?9. Make f(n) function = 7 log n + 2. Therefore, prove that this function is f(n) = O(log n) and also the function type.10. Make f(n) = 8n log n-2. Therefore, prove that this function is f(n) = ð(n log n) and identify the function type.11. Make it f(n) = 8n + 12. Therefore, prove that this function is f(n) = ş(n) and identification of the type of function.12. Make it f(n) = 3n log n + 2n. So prove that this function f(n) = ş(n log n) and identify the type of function.iii12 zThe basics of algorithms algorithms

artist cv template for mac , 5191733.pdf , vajokugeguzozula.pdf , graphic design apps free , kido group corporation annual report , timixe_kevomoseburabe.pdf , teamviewer 10 license key free download , wemewefe_vavikelakatotak_bewawoxuv.pdf , ibm aix commands pdf free download , chopin classical music , regoposoponut.pdf , plant kingdom notes for neet pdf , wirokuwofubonoderax.pdf ,