Algorithms For Perfect Squares
Lesson Plan: Class 08 / ALG / 05

cs pathshala
Bringing computational thinking to schools

Overall goal of the lesson	Students learn about writing Algorithmic steps for simple procedures
Prior Knowledge Required	Square and Square Roots

Goal:	Build an algorithm that can be used to determine whether a given number is a perfect square
Description:	Given any input number, determine if the number is a perfect square.
Material Required:	1. One copy of worksheet per student 2. Pen/Pencil, eraser. 3. Lesson Presentation
Procedure Summary:	1. Go through the presentation. 2. Follow the algorithm in the presentation with examples 3. Make sure every student is following the slides. 4. Provide the worksheets, once presentation is done. 5. Let the students solve them and help them with answers.
Procedure Details:	**Slide 1:** Title Slide **Slide 2:** Discuss with the students about Perfect Squares and their properties. Discuss about how the properties can be checked on large numbers, very quickly. If required, let the students write new algorithms based on properties they came up with earlier. **Slide 3-4:** Take few simple numbers - square and non-square numbers - and ask the students to find out whether they are perfect squares or not. Let the students go through whatever methods they already know to check if they are perfect squares. In Slide 4, ask the students to determine how and why the flowchart is terminated. **Slide 5:** Introduce the students to a rote procedure to check whether a number is a perfect square. The slide contains only the flowchart. Let the students spend some time on the flowchart – identifying the decision points, and the repeatable steps in the flow chart. Allow the students to look at each step and understand what checks are being done and why are they necessary. **Slide 6:** This is a detailed flowchart for the steps in the algorithm. Talk to the students about the number of different calculations (like additions and multiplications) that are represented in the steps of the flowchart. Also make the students guess the number of such steps needed in a large number. Talk to the students about the relation between the numbers of operations performed by any algorithm for a given input number. The behavior of different algorithms can be compared based on the operations performed for the same input. Let the students also know the different variables being used in the algorithm, and how they help in determining the answer.

	Slide 7: Discuss with the students about ways to improve the algorithm. Any idea to quickly terminate the loop for a non-square or to skip a particular number is to be discussed. Discuss how we can start our algorithm with an approximate guess to multiply and compare with the input number. **Slide 8:** Talk to the students about the sequence of perfect squares. What sequence to they follow? Talk about the sum of sequential odd numbers leading to a perfect square as an example. Draw a square of n x n, and (n + 1) x (n + 1). What is the difference between them? Talk to the students about the relation between the sum of the first n odd-numbers and the square of the number n. (Writing an algorithm for this is also given as an exercise in the worksheet for the lesson). **Slide 9:** Talk to the students about using other known algorithms to find if a number is a perfect square. The students may already know about prime factors for a number. Now, let them think about using the prime factors of a number, and to determine whether a number is a perfect square from that information. Ask the students about Perfect Cubes? How are the prime factors for a Perfect Cube similar to those of a Perfect Square? **Slide 10:** Given this algorithm, let the students discuss if the same can be used to find the square root of a perfect square? The same is given in the worksheet as well. This exercise can also be given as a class assignment [15 mins]. You can also talk about the Newton-Rhapson(N-R) method for finding the square root for any non-square number as well, using approximation as a technique. In N-R method for finding square roots, we can use the below formula to find square root of any number to the desired precision. $\sqrt{S} \approx \{X_{n+1} = \frac{1}{2} (X_n + S/X_n)\}$ For e.g., $\sqrt{5}$ can be calculated as below. Let us start with a guess of 2. So, $X_0 = 2$. $X_1 = \frac{1}{2} (2 + 5/2) = 2.25$. $X_2 = \frac{1}{2} (2.25 + 5/2.25) = 2.236$, and so on, up to the desired precision. Please note that the method can also be generalized to find roots of any equation. You may introduce the students to really large numbers that are thousands of digits long and the complexities when applying the same algorithm. The same algorithm may not work efficiently in such cases, and there are other shortcuts to arrive at the result. There are some quick checks that can determine if a number is not a perfect square. For e.g., What are the possible combinations of the last digits for perfect squares? Numbers that end with 2, 3, 7 and 8 cannot be perfect squares. Similarly numbers that end with odd numbers of 0s cannot be perfect squares. Perfect squares that end with 5, always have 2 in the ten's digit. **Slide 11:** Thank you.