Lesson 7: Merge Sort

In this lesson, students will learn about the merge sort algorithm, which is a divide-and-conquer algorithm used for sorting large lists of data efficiently. The lesson will begin with a discussion on how to combine two groups of sorted items into one sorted group. Then, students will be introduced to the step-by-step visualization of the merge sort algorithm through slides or visual aids. They will learn how the algorithm splits the list into smaller sublists, sorts those sublists, and merges them back together. During guided practice, students will have the opportunity to practice executing a full merge sort on provided sample data. They will also be assigned additional practice problems to complete independently. The lesson will conclude with an exit ticket to assess students' comprehension of the merge sort algorithm and a recap of the main points covered.

## **Objectives:**

- Merge two ordered lists of items into a new ordered list.

- Describe how merge sort is used for ordering a list of items.

- Perform a merge sort to order a list containing sample data.

## **Materials:**

- Slides or visual aids

- Worksheets with sample data

- Pen or pencil

## **Bell-Ringer Activity:**

- Ask students to think about how they would combine two groups of sorted items into one sorted group.

- Give them a few minutes to brainstorm and then ask for volunteers to share their ideas.

## **Introduction:**

- Explain to students that in this lesson, they will be learning about the final sorting algorithm in the unit, merge sort.

- Tell them that merge sort is a divide-and-conquer algorithm that works by repeatedly dividing the list into smaller sublists, sorting those sublists, and then merging them back together.

- Emphasize that merge sort is an efficient algorithm for sorting large lists of data.

## **Direct Instruction:**

- Present slides or visual aids that provide a step-by-step visualization of the merge sort algorithm.

- Start by explaining how the algorithm splits the list until each item is in a list of its own.

- Then, demonstrate how pairs of lists are merged together in order.

- Highlight the importance of identifying which pairs of lists should be merged together and which items in a pair of lists need to be compared at each stage in the algorithm.

## **Guided Practice:**

- Distribute worksheets with sample data to the students.

- Instruct them to practice executing a full merge sort on the provided samples.

- Remind them to refer to the partially worked examples on the worksheets for guidance.

- Circulate the classroom to provide assistance and answer any questions.

## **Independent Practice:**

- Assign additional practice problems for students to complete on their own.

- Encourage them to apply the merge sort algorithm to order a list containing sample data.

- Collect and review their completed assignments to assess their understanding.

## **Exit Ticket:**

- Administer an exit ticket to assess students' comprehension of the merge sort algorithm.

- Ask them to describe the steps involved in merge sort and explain why it is an efficient sorting algorithm.

- Collect the exit tickets before the end of the class.

## **Closure:**

- Recap the main points covered in the lesson, emphasizing the objectives achieved.

- Remind students of the importance of merge sort in efficiently sorting large lists of data.

- Encourage them to continue practicing and applying the merge sort algorithm in their future studies.

## **Common Core Standards:**

- CCSS.ELA-LITERACY.RST.9-10.3: Follow precisely a complex multistep procedure when carrying out experiments, taking measurements, or performing technical tasks, attending to special cases or exceptions defined in the text.

- CCSS.ELA-LITERACY.RST.9-10.4: Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 9-10 texts and topics.