

# Recursion

Author: Keith Alcock 02/16/2007 09:26:00 AM MDT

## VITAL INFORMATION

---

<b>Subject(s):</b>	Computer Science, Java programming
<b>Topic or Unit of Study:</b>	Recursion
<b>Grade/Level:</b>	9-12
<b>Objective:</b>	At the conclusion of this lesson students will be able to:  <ol style="list-style-type: none"><li>1. define recursion.</li><li>2. recognize recursion in a program.</li><li>3. convert simple iteration into simple recursion</li><li>4. write recursive code to solve a simple problem.</li><li>5. identify typical recursion strategies such as<ol style="list-style-type: none"><li>a) helper functions,</li><li>b) state-holding arguments,</li><li>c) and counting down.</li></ol></li><li>6. defend the assertion that recursion and iteration are equivalent.</li><li>7. apply the recursive technique to a suitable problem.</li></ol>
<b>Summary:</b>	Students learn about recursion based on their previous knowledge of iteration.

## IMPLEMENTATION

---

<b>Learning Context:</b>	Students have studied iteration extensively, but also need to know about recursion. They have just worked the Fibonacci numbers problem iteratively even though the simpler solution is recursive.	
<b>Procedure:</b>	I scripted and rehearsed this lesson and prepared all code. My rehearsal notes are attached as well as three source code files. One file contains iterative versions of all the code, while another contains the recursive versions. The third file shows only the method signatures of the recursive versions and was used for note taking and review. It also describes problems that the students will work on independently.  <b>Attachments</b> <table border="1"><tr><td><ol style="list-style-type: none"><li>1. <a href="#">Recursion.doc</a></li><li>2. <a href="#">ReflectiveTeachingLog.doc</a></li></ol></td></tr></table>	<ol style="list-style-type: none"><li>1. <a href="#">Recursion.doc</a></li><li>2. <a href="#">ReflectiveTeachingLog.doc</a></li></ol>
<ol style="list-style-type: none"><li>1. <a href="#">Recursion.doc</a></li><li>2. <a href="#">ReflectiveTeachingLog.doc</a></li></ol>		
<b>Differentiated Instruction:</b>	For the particular target audience, no differentiated instruction is anticipated, although plenty of opportunities exist.	
<b>Sample Student Products:</b>	Students are to enter the code we developed at the board (and they may have recorded on their papers) to verify that it works and to	

review the material. This product should be evaluated.

**Collaboration:** Students will work collaboratively & individually. Students will work in groups of 2.

**Time Allotment:** 2 class periods. 1.25 Hrs per class.

**Author's Comments & Reflections:** A reflective teaching log is attached to the procedure section, since it cannot be attached here.

## **MATERIALS AND RESOURCES**

---

**Instructional Materials:** Three source code files are attached.

### **Attachments**

1. [Iteration.java](#)
2. [Recursion.java](#)
3. [RecursionCode.java](#)

**Resources:**

- Technology resources:  
Java compiler
- The number of computers required is 1 per student.

## **STANDARDS & ASSESSMENT**

---

**Standards:**

**Assessment/Rubrics:** Students should be able to recode everything that was developed at the board. Several problems are available for the students to complete on their own. Assessment can be based on the number and/or correctness of the solutions.