

## Homework 5: Stacks

cs230  
Spring 2002

Allen B. Downey  
Computer Science Department

Due: Thursday 14 March

The purpose of this assignment is to continue our work with lists and start in on Stacks.

### Generic lists

Look over your solution to Homework 4 and my solutions from the web page:

<http://rocky.wellesley.edu/cs230/solutions/hw04>

Patch up any errors in your solution and continue the process of cleaning up your code and making it more presentable.

1. Transform your list implementation into a generic list by making the cargo an `Object` instead of an integer.
2. Write some client code that tests your new and improved implementation.

### Implement `split`

1. Write a `LinkedList` method called `split` that takes a `String`, breaks it up into words (using spaces as delimiters), and returns a list of `Strings`, with one word per list node.

Your implementation should be efficient, meaning that it takes time proportional to the number of words in the string.

### Check balance and nesting

1. Write a program called `Balance.java` that reads a file and checks that the parentheses `()` and brackets `[]` and squiggly-braces `{}` are balanced and nested correctly.

You can use the file I/O code from Homework 0, and you can use Java's built-in implementation of a `Stack`.

### Implement your own stack

1. Write an implementation of the `Stack` ADT using your list implementation as the underlying data structure.
2. In the previous program (`Balance`), replace the built-in stack with your implementation and make sure that the program still works. You should not have to make any changes in `Balance.java` when you switch implementations.