Project 4. Sharing

In directory `/comp/150C++/project4/` are some source files that implement the “rope” data type of Projects 2 and 3. The approach from Project 3 has been revised slightly by nesting the body of `implementation.h` into the `Rope` class. Now `rope.h` is the only header file, and it contains only declarations, except for the body of the default constructor for `Rope`.

Your task in this project is to use reference counting to control the deletion of objects. You are to provide, in a file `rope.cc`, the bodies of all methods declared but not defined in `rope.h`, and to maintain and use the declared reference counters properly. You can simply copy `rope1.cc` and your `rope2.cc` from Project 3 and modify them to support reference counting. You will also have to add the qualifier `Rope::` to many of the existing references to `Implementation` and its derived classes and methods.

Instead of using your files from Project 3, you may start from the provided file `basic-rope.cc` which lacks just reference counting and deleting of objects.

Store your code in a file “rope.cc” and use the supplied Makefile to compile it into one or more test programs. You can probably test and debug most effectively by writing your own mainlines, in addition to the ones provided. Although you will not need to write very many new lines of code in `rope.cc`, it may be tricky to get them exactly right; furthermore, when they fail, the symptoms may be mysterious. Thus the debugging of this project is likely to be more difficult than before.

You should stick to the approach, using virtual methods, etc., that was used in Project 3. As always, you are expected to produce code that is completely correct.

Submit your file by Tuesday, April 6 at 5:00 p.m. using this command:

```
provide 150C++ project4 rope.cc
```