Assignment: DNA Splicing

Restriction Enzyme Cleaving Explained

You'll need to understand the basic idea behind restriction enzymes to understand the simulation you'll be writing.

Restriction enzymes cut a strand of DNA at a specific location, the binding site, typically separating the DNA strand into two pieces. Given a strand of DNA "aatccgaattctac" and a restriction enzyme like EcoRI "gaattc", the restriction enzyme locates each occurrence of its pattern in the DNA strand and divides the strand into two pieces at that point, leaving either blunt or sticky ends as described below. In the simulation there's no difference between a blunt and sticky end, and we'll use a single strand of DNA in the simulation rather than the double-helix/double-strand that's found in the physical/real process.

Restriction enzymes have two properties or features: the pattern of DNA that marks a site at which separation occurs and a number/index that indicates how many characters/nucleotides of the pattern attach to the left-part of the split strand. For example, the above diagram shows a strand split by EcoRI. The pattern for EcoRI is "gaattc" and the index of the split is one indicating that the first nucleotide/character of the restriction enzyme adheres to the left part of the split.

In some experiments, and in the simulation you'll run, another strand of DNA will be spliced into the separated strand. The strand spliced in matches the separated strand at each end as shown in the diagram below where the spliced-in strand matches with G on the left and AATTTC on the right as you view the strands.

When the spliced-in strand joins the split strand we see a new, recombinant strand of DNA as shown below. The shaded areas indicate where the original strand was cleaved/cut by the restriction enzyme.
Your code will be a software simulation of this recombinant process: the restriction enzyme will cut a strand of DNA and new DNA will be spliced in to create a recombinant strand of DNA. In the simulation the code changes every occurrence of the restriction enzyme to include the splicee.

Here is a test that generates the sequence above by cutting and splicing in the sequence of "tgata" once with 1 nucleotide 'G' to the left of the splicee “tgata”.

```java
@Test
public void showStrandCutAndSpliceFromSpec() {
    LinkedStrand strand = new LinkedStrand(1, "gaattc");
    assertEquals(1, strand.getIndex());
    assertEquals("gaattc", strand.getRestrictionEnyme());

    strand.append("aatccgaattctatc");
    assertEquals(16, strand.size());
    assertEquals("aatccgaattctatc", strand.getStrand());

    strand.cutAndSplice("tgata");
    assertEquals(21, strand.size());
    // New DNA material here
    //                  |||||
    assertEquals("aatccgtgataaattctatc", strand.getStrand());
}
```

And here is `interface DNASTrand` that must be implemented by class `LinkedStrand`

```java
public interface DNASTrand {
    /**
     * Return the number of nucleotides in this strand.
     */
    public long size();

    /**
     * Return the pattern that will be search for when splicing
     */
    public String getRestrictionEnyme();

    /**
     * Return the number of nucleotides in the strand as a String
     */
    public String getStrand();

    /**
     * The number of nucleotides from the restriction enzyme that stay to the "left"
     */
    public int getIndex();

    /**
     * Add all nucleotides in str to the end of this LinkedStrand.
     * This must run O(1) by maintaining a reference to the last node.
     */
    public void append(String str);

    /**
     * Perform a cut at each occurrence of restrictionEnyme and insert the nucleotides
     * simulated by the String toBeSpliced. There will be getIndex() nucleotides of the
     * restrictionEnyme to the "left" and the other nucleotides on the right side.
     */
}
```
public void cutAndSplice(String toBeSpliced);

Implementation Specifics

You'll be developing/coding a class LinkedStrand that implements a Java interface DNASTrand. The class simulates cutting a strand of DNA by a restriction enzyme and appending/splicing-in a new strand.

You must use a singly linked structure to support the operations—specifically the class LinkedStrand must maintain references to a singly linked structure containing characters to represent a DNA strand. You must keep and maintain a reference to the first Node of the linked list and to the last node of the linked list. These references are maintained as class invariants -- the property of pointing to first/last nodes must hold after any method in the class executes (and thus before any method in the class executes).

TODO: DRAW A LINKED LIST

The diagram below shows the results of cutting an original strand of DNA at three points and then splicing-in the strand "GTGATAATTTC" at each of the locations at which the original strand was cut. Since splicing into a linked list is a constant-time, O(1) operation this implementation should be more efficient in time and space when compared to the String implementation.
Linked List Details

- Store nucleotides as single character data in a singly linked structure of Node objects
- Use a dummy header that stores a character that is not part of the strand. This allows appends of new nucleotides at the end without checking to see if first is null each time (no special case needed).
- New nucleotides are appended at the "end" of the DNASTrand in the append method

OPTIONAL READING

Background on DNA, Restriction Enzymes, and PCR

Restriction Enzymes
(source: http://www.astbury.leeds.ac.uk/gallery/leedspix.html)
This background is interesting, but not really needed to do the assignment. There are some good stories here, but if you want to get to the assignment, you can skip this stuff.

Three scientists shared the Nobel Prize in 1978 for the discover of restriction enzymes. They're also an essential part of the process called PCR polymerase chain reaction which is one of the most significant discoveries/inventions in chemistry and for which Kary Mullis won the Nobel Prize in 1993.

Kary Mullis, the inventor of PCR, is an interesting character. To see more about him see this archived copy of a 1992 interview in Omni Magazine, this 1994 interview as part of virus myth, his personal website which includes information about his autobiography Dancing Naked in the Mind Field, though you can read this free Nobel autobiography as well.

You can see animations and explanations of both restriction enzymes and PCR at DnaTube and Cold Spring Harbor Dolan DNA Learning Center.
The simulation is a simplification of the chemical process, but provides an example of the utility of linked lists in implementing a data structure. The linked list code you'll write and reason about is an example of a chunk list. You can do more work with chunk lists for extra credit.

**Genesis of Assignments Linked to DNA/Genomics**

TBA: Credit Rich Pattis (Carnegie Mellon) and Owen Astrachan (Duke)

**Grading Criteria**

- 100pts WebCat Code and Problem Coverage

- 80 If you did not use a singly linked structure

- 20 If you did not use a singly linked structure but did not employ last to allow O(1) appends

- 20 If you cannot cutAndSplice O(n). Make sure you go through the strand only once.

  Use the references to the first and last nodes to insert the new DNA material