Introduction

As we know from the reading assignment, linked lists are data structures with elements organized sequentially, and joined together with links. These elements are placed along with their links that point to the following element in what are referred to as “nodes”. These nodes are usually visually illustrated as boxes, “with the element written inside the box and the link drawn as an arrow, pointing out of the box” (Main, 2011, p. 176). The arrows are then used to bridge the nodes to each other.

Uses

Since I work in the bioinformatics field, I will be briefly describing how the linked list data structure can be used for bioinformatics solutions. DNA sequences are composed of four different nucleotides (bases): adenine, guanine, cytosine, and thymine. Most of the time, these bases are simply represented by using their first letter (AGCT). DNA contains genetic information, in which 3 nucleotides form what is called a codon. These codons encode specific amino acids, depending on their contents and sequence of nucleotides. For example, the codon sequence CTA encodes for the amino acid Leucine, and the codon sequence ACT encodes for the amino acid Threonine (Nature Education, 2013). I will not get into the specific details of the biochemistry and genetics of this because to do so would require too much space, so those interested can do a Google search on this topic.

Since DNA is sequential, the linked list is an ideal data structure to represent it. The first example I will use involves the organism ciliates, which are single-celled organisms, and have been studied in part because of their “unusual way of organizing their genome” (Harju et al, 2007). During reproduction, the organisms actually splice genetic material out of their chromosomes, which are then assembled together to form germline DNA. Ciliates are able to correctly identify which genes to splice from their chromosomes by arranging “their genomic data as linked lists” (Harju et al, 2007). Each gene spliced contains the DNA sequences that identify their start and end, much like linked lists. Thus, the use of this data structure along with complex permutations (which are beyond the scope of this course) makes it possible to better predict gene assembly.

My second example also involves the use of the linked list in simulating the organization of DNA sequences. The use of the linked list data structure is partially responsible for the completion of the human genome project. As DNA is sequenced, it is not perfect, and thus a quality score is assigned to each base (Li and Khuri, 2004). As these fragments are assembled, a linked list can be used to store fragment ID, as well as their quality scores. The linked lists can then be sorted based on this score as the structure is traversed. Nodes can then be removed and reinserted accordingly. Just like the previous example, the way in which these DNA fragments are assembled is a “very complex problem in computational biology” (Li and Khuri, 2004), which would be very difficult to explain.

Future of linked lists in bioinformatics

The linked list data structure will continue to be useful in the bioinformatics field. Even though the human genome project is complete, there are many other genomes that are not. In the future, newer and faster algorithms may be developed, which can more accurately analyze genetic data.

References:

Main, M. (2011) *Data Structures and Other Objects Using Java* (4th Ed.) Prentice Hall. [Kindle Edition], p. 176.

Nature Education (2013) *The Information in DNA Determines Cellular Function via Translation* [Online]. Available from: <http://www.nature.com/scitable/topicpage/the-information-in-dna-determines-cellular-function-6523228> (Accessed: 11 August 2013)

Harju, T., Li, C., Petre, I. & Rozenberg, G. (2007) ‘Complexity Measures for Gene Assembly’, *Knowledge Discovery and Emergent Complexity in Bioinformatics Springer*, pp. 42-60.

Li, L. & Khuri, S. (2004) ‘A Comparison of DNA Fragment Assembly Algorithms.’, *METMBS*, pp. 329.