List Observation Report

Dr. Rouse

Ancel Carson

9/21/2018

Differences in Data types

Array: An array is the simplest of the data types. It is easy to index inputs in an ordered fashion in the computer’s memory. All the data is stored together one after the other and can easily be accessed in any order. The main drawback of an array is that the size of the array needs to be set early and it cannot be changed later in the program.

ArrayList: An ArrayList functions internally much like an array. Information is kept together in a row and theoretically could be quickly accessed. The main difference between an array and an ArrayList is that the ArrayList does not have a fixed initial size. As more data is added the length of the list gets longer. Another difference between the two is that the numbers aren’t indexed. They are given a position from the front of the list. This allows the programmer to add or remove data in different locations and keep the order sequential.

LinkedList: A LinkedList functions like an ArrayList in that data can be added and removed in any order as well as not being limited to a set amount. Where it differs from an array is that the data is not stored in a line like an array is. A LinkedList functions using nodes with data and references. The LinkedList creates a new item every time something is added to it. These items do not need to be in a line or even in a pattern as each node has a reference to the next node. Starting from the beginning, the list is worked through going from reference to reference. A potential drawback from this list type is that if a reference is somehow broken all following nodes will be lost.

BackwardArrayList: A BackwardArrayList is best described as the mirror to an ArrayList. The information is stored at the end of an array where the last position of the array is the first position of the list. Input is also added backwards from an ArrayList. When a new item is added, all of the information is shifted one space over to make room for the new data point.

Results

The first three: array, ArrayList, and LinkedList all took around 2-3 seconds to complete adding the 2.5 million integers. The BackwardArrayList took around 24 minutes and 33 seconds to complete on its best time. To test the effects of the computation needed I ran the program 8 times at once to see how long it would take. The final time was recorded at 132 minutes and 42 seconds. The reason the BackwardArrayList took so much longer is that for each integer added all other integers had to shift down one space to make room at the beginning of the list. When there were 24999999 integers they all had to shift over to make room for the final integer reaching the 2.5 million total.

Data Structure for Task

Depending on a few things the different structures could each be the right choice for the task. If the initial length of the file known, the array is best suited because it is the simplest and easiest to retrieve data from. If memory space is not an issue, the ArrayList might be the best option. You do not need the initial size to receive all the data and it is all kept close together. There will be empty spaces at the end of the list that take up memory but if there is enough memory there is not an issue. If memory space is limited, the LinkedList is the better choice. It does not have a set size and there are only as many items as there are data points.

Potential Tasks

Array: The best task for an array is to hold a known amount of data that needs to be drawn from in any order and quickly manipulated.

ArrayList: An ArrayList would be used when the initial size of the array is not known and information will most likely be added to the end. It would be a good thing to have in a sorting algorithm because data can easily be moved around.

LinkedList: The best task for a LinkedList would be one where there is a limited amount of memory. This could either be caused by a computer that doesn’t have much memory or a program there empty spaces are not things it can afford. This would be something like count of data points from an ongoing system.

BackwardArrayList: A BackwardArrayList would be used if data collection needs to be in reverse order from how it was entered. An example you be keeping a log and needing the first thing that comes up to be the most recent thing that happened.