

## Activity about singly linked list

### 1.What is a singly linked list, and how does it differ from an array?

A singly linked list is a linear data structure in which each element, called a node, contains two parts “the data and a reference” (or pointer) to the next node in the sequence. The list starts at the head node, and each node links to the next until the end of the list (where the pointer is null).

Unlike arrays, linked lists do not store elements in contiguous memory locations. Arrays have fixed sizes (in most languages) and allow random access to elements via an index, while linked lists have dynamic sizes and require sequential traversal to access elements.

### 2.When would you prefer a linked list over an array, and vice versa?

For me, I'd choose a linked list when I know I'll be doing a lot of insertions or deletions, especially in the middle of the data. It's also handy if I'm not sure how big the list will get, since linked lists can grow and shrink without worrying about running out of space like an array might.

On the other hand, I'd pick an array when I need to quickly access elements by their position, or if I already know the size of the data in advance.

### 3.How are linked lists used in real-world applications?

In real life, linked lists quietly work behind the scenes in many of the tools we use every day. For example, when you're browsing the internet, your browser history is often managed like a linked list, letting you move back and forth between pages you've visited. In text editors, the undo and redo features rely on a similar structure, keeping track of each change so you can step backward or forward through your work. Even things like music playlists or certain memory management systems use linked lists to make it easy to add, remove, or rearrange items without a lot of extra work.

### 4.References

Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). Data Structures and Algorithms in Java (6th ed.). Wiley.

GeeksforGeeks. (n.d.). Singly Linked List. Retrieved from <https://www.geeksforgeeks.org/singly-linked-list>

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.