Question 1

1. Data type:
- char* pStr points to a character.
- int* pArr points to an integer.
2. Memory Usage:
- Each element pointed to by char* pStr typically uses 1 byte.
- Each element pointed to by int* pArr typically uses 4 bytes (this can vary depending on the system).
3. Common Usage:
- char* pStr is commonly used to handle strings (null-terminated sequences of characters).
- int* pArr is commonly used to handle arrays of integers.

b. What are the advantages and disadvantages of a singly linked list compared to the array?

1. Advantages:
- Dynamic Size: A singly linked list can easily grow and shrink in size.
- Ease of Insertion/Deletion: Inserting or deleting an element in a singly linked list does not require shifting elements as in an array (O(1) for insertion/deletion at the beginning).
2. Disadvantages:
- Memory Overhead: Each element in a singly linked list requires extra memory for the pointer to the next element, increasing the overall memory usage compared to an array.
- Access Time: Accessing an element in a singly linked list requires traversing from the head node to the desired node, resulting in O(n) time complexity, whereas array elements can be accessed directly using an index in O(1) time.

c. Can we apply a binary search approach to find a number in an ordered linked list? Why?

No, we cannot efficiently apply a binary search approach to find a number in an ordered linked list. The binary search algorithm requires random access to the middle element of the collection to divide the search space in half.