# Linked List – Level Beginner

1. Explain the trade-offs between the linked list data structure and the array data structure.

Complete the following questions about the linked list using the Node structure shown below. You may use the printList function below to verify your answer.

|  |
| --- |
| struct Node {  double num;  Node \*next;  };  void printList(Node \*head)  {  while (head != 0)  {  cout << head->num << " ";  head = head->next;  }  cout << endl;  } |

Table 1: Node Structure and PrintList Function

|  |
| --- |
| int main()  {  Node \*head = new Node;  head->num = 3.4;  head->next = new Node;  Node \*curNode = head->next;  curNode->num = 2.2;  curNode->next = 0;  printList(head);  return 0;  } |

Table 2: Example of printList Function Call

1. Below is some code implementing a linked list of size two. Modify the code so that an additional node is added to the beginning of the list whose value is 5.4. See the diagram included below for reference.

|  |
| --- |
| From:    To: |

|  |
| --- |
| int main()  {  Node \*head = new Node;  head->num = 3.4;  head->next = new Node;  Node \*curNode = head->next;  curNode->num = 2.2;  curNode->next = 0;  // Insert Code Here  return 0;  } |

1. Below is some code implementing a linked list of size two. Modify the code so that an additional node is added to the **middle** of the list whose value is 5.4. See the diagram included below for reference.

|  |
| --- |
| From:    To: |

|  |
| --- |
| int main()  {  Node \*head = new Node;  head->num = 3.4;  head->next = new Node;  Node \*curNode = head->next;  curNode->num = 2.2;  curNode->next = 0;  // Insert Code Here  return 0;  } |

1. Below is some code implementing a linked list of size two. Add additional code so that the first node (one whose value is 3.4) in the list is removed. Do this without modifying any code above the comment.

|  |
| --- |
| From:    To: |

|  |
| --- |
| int main()  {  Node \*head = new Node;  head->num = 3.4;  head->next = new Node;  Node \*curNode = head->next;  curNode->num = 2.2;  curNode->next = 0;  // Insert Code Here  return 0;  } |