**Doppelt verkettete Listen**

|  |
| --- |
| Eine **Liste** (engl. list) ist eine lineare Datenstruktur, die beliebig viele Objekte verwalten kann. Das Einfügen und Löschen von Elementen ist an jeder Position der Liste möglich. |

**Das Anhängen (append)**

**Fall 1: Die Liste ist leer.**

|  |  |  |
| --- | --- | --- |
| null  null  null  list  first =  last =  current = | node1  nextNode =  prevNode =  object =  list  first =  last =  current = | **Schritt 1:**  **Schritt 2:** |

**Fall 2: In der Liste ist mind. ein Element.**

|  |  |  |
| --- | --- | --- |
| node1  nextNode =  prevNode =  object =  list  first =  last =  current = | node2  nextNode =  prevNode =  object =  node1  nextNode =  prevNode =  object =  list  first =  last =  current = | **Schritt 1:**  **Schritt 2:**  **Schritt 3:**  **Schritt 4:** |

**Das Einfügen an der aktuellen Stelle (insert)**

**Fall 1: Die Liste ist leer.**

|  |  |
| --- | --- |
| Was ist in diesem Fall zu tun? | **Schritt 1:** |

**Fall 2: Das aktuelle Element ist null.**

|  |  |
| --- | --- |
| Was ist in diesem Fall zu tun? | **Schritt 1:** |

**Fall 3: Das aktuelle Element ist das letzte.**

|  |  |  |
| --- | --- | --- |
| node1  nextNode =  prevNode =  object =  list  first =  last =  current = | node2  nextNode =  prevNode =  object =  node1  nextNode =  prevNode =  object =  list  first =  last =  current = | **Schritt 1-4:**  **Schritt 5:** |

**Fall 4: Alle anderen Fälle**

|  |  |
| --- | --- |
| node2  nextNode =  prevNode =  object =  node1  nextNode =  prevNode =  object =  list  first =  last =  current = | **Schritt 1:**  **Schritt 2:**  **Schritt 3:**  **Schritt 4:**  **Schritt 5:**  **Schritt 6:** |
| node2  nextNode =  prevNode =  object =  node3  nextNode =  prevNode =  object =  list  first =  last =  current =  node1  nextNode =  prevNode =  object = |

**Das Löschen des aktuellen Knotens (remove)**

**Fall 1 + 2: Die Liste ist leer. / Das aktuelle Element ist null.**

|  |  |
| --- | --- |
| Was ist in diesem Fall zu tun? | **Schritt 1:** |

**Fall 3: Der aktuelle Knoten ist der einzige.**

|  |  |
| --- | --- |
| Was ist in diesem Fall zu tun? | **Schritt 1-3:** |

**Fall 4: Der aktuelle Knoten ist der erste.**

|  |  |  |
| --- | --- | --- |
| node1  nextNode =  prevNode =  object =  node2  nextNode =  prevNode =  object =  list  first =  last =  current = | node2  nextNode =  prevNode =  object =  list  first =  last =  current = | **Schritt 1:**  **Schritt 2:**  **Schritt 3** |

**Fall 5: Der aktuelle Knoten ist der letzte**

|  |  |  |
| --- | --- | --- |
| node2  nextNode =  prevNode =  object =  node1  nextNode =  prevNode =  object =  list  first =  last =  current = | null  node1  nextNode =  prevNode =  object =  list  first =  last =  current = | **Schritt 1:**  **Schritt 2:**  **Schritt 3** |

**Fall 6: Alle anderen Fälle**

|  |  |
| --- | --- |
| node3  nextNode =  prevNode =  object =  node2  nextNode =  prevNode =  object =  list  first =  last =  current =  node1  nextNode =  prevNode =  object = | **Schritt 1:**  **Schritt 2:**  **Schritt 3** |