

Wintersemester 2014/2015  
**Übungen zur Vorlesung**  
**Algorithmisches Denken und imperative Programmierung (BA-INF-014)**  
**Aufgabenblatt 6**  
Zu bearbeiten bis: 06.12.2014

**Aufgabe 1** (*Listen - 16 Punkte*)

Sie haben in der Vorlesung die Datenstruktur *IntNode* für Listen über *Integer* Zahlen sowie Funktionen für die Listenoperationen kennengelernt.

a) Sei folgende main-Funktion gegeben:

```
int main(int argc, char *argv[])
{
    IntNode *L2, *L1 = NULL;
    L1= insertFirst( L1, 3);
    L1= insertFirst( L1, 7);
    L1= insertFirst( L1, 11);
    L2=L1;
    L2= insertFirst( L2, 5);
    L2= insertFirst( L2, 14);

    \\ Stelle1

    printList(L1);
    printList(L2);
    return 0;
}
```

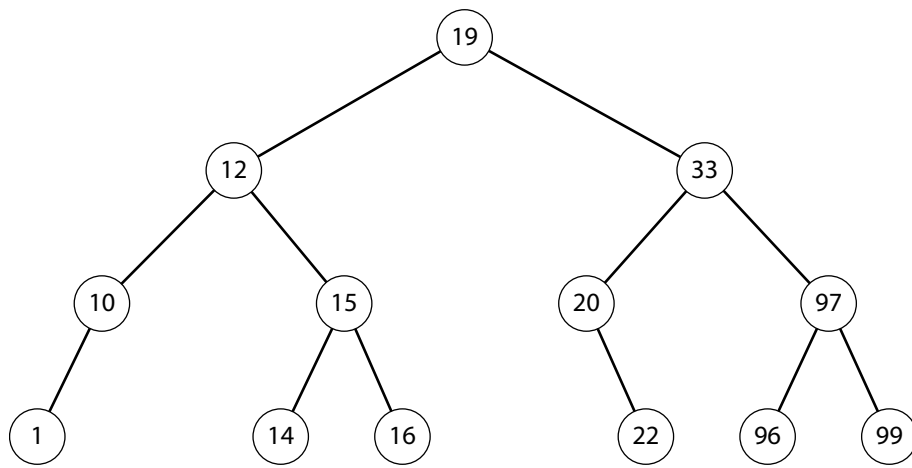
Skizzieren Sie den Zustand des Speichers an **Stelle 1**.

b)

- a) Implementieren Sie entsprechende Datenstruktur *DoubleNode* für Listen über *Double* Zahlen.
- b) Implementieren Sie die Funktion *DoubleNode \* insertFirst( DoubleNode \*head, double c)* zum Einfügen von *double d* am Anfang einer Liste.
- c) Implementieren Sie die Prozedur *void printList(DoubleNode \*head)*, die den Inhalt einer Liste auf der Konsole ausgibt.
- d) Implementieren Sie die Funktion *DoubleNode \* insertLast( DoubleNode \*head, double d)* zum Einfügen von *double d* am Ende einer Liste.
- e) Implementieren Sie die Funktion *DoubleNode \* reverseDoubleListCon( DoubleNode \*head)* zum konstruktiven Invertieren einer Liste.
- f) Implementieren Sie die Funktion *DoubleNode \* reverseDoubleList( DoubleNode \*head)* zum destruktiven Invertieren einer Liste.

**Aufgabe 2** (*Baumdurchläufe - 4 Punkte*)

Gegeben sei folgender Binärbaum:



a) Geben Sie die Elemente des Baumes in der Reihenfolge an, in der sie bei einem *preorder*-Durchlauf bearbeitet werden!

b) Geben Sie die Elemente des Baumes in der Reihenfolge an, in der sie bei einem *postorder*-Durchlauf bearbeitet werden!

c) Geben Sie die Elemente des Baumes in der Reihenfolge an, in der sie bei einem *inorder*-Durchlauf bearbeitet werden!