

## 9. Übungsaufgaben zur LV Algorithmen & Datenstrukturen

Abgabetermin: Mi, 26.05.04

### Aufgabe 1:

- Implementation der Methode *entfernen()* aus der Klasse *Segmentbaum* :

```
private boolean entfernen(LIntervall iv, int i) {
    boolean ok=false;
    if(knInt(i).istIn(iv))
        ok=((AVLBaum) getElement(i)).entfernen(iv.getSchluessel());
    else {
        int j=2*i+1;
        if(istPos(j) && knInt(j).hatSchnitt(iv)) ok=entfernen(iv,j);
        if(istPos(j+1) && knInt(j+1).hatSchnitt(iv)) ok=(entfernen(iv,j+1)||ok);
    }
    return ok;
}
```

### Aufgabe 2:

- Implementation der Methode *report()* aus der Klasse *Segmentbaum* :

```
private void report(int i, int nadel, AVLBaum Schnittmenge) {
    if(((AVLBaum) getElement(i)).Wurzel!=null) {
        AVLKnoten SchlKn=((AVLBaum) getElement(i)).minSchlKn();
        do{
            Schnittmenge.einfuegen(SchlKn.getInhalt());
            SchlKn=((AVLBaum) getElement(i)).nextSchl(SchlKn);
        }while(SchlKn!=null);
    }
    int j=2*i+1;
    if(istPos(j) && knInt(j).enthaelt(nadel)) report(j, nadel, Schnittmenge);
    else if(istPos(j+1)) report(j+1, nadel, Schnittmenge);
}
```

**Beispielausgabe:** Intervalle im Segmentbaum: [1,3), [0,3), [1,7), [2,6), [4,5), [5,8), [3,8), [4,7)

