

Opgave 1

- a) Indien in een public class een object geïmplementeerd wordt waarvan het de bedoeling is dat dit object in principe in ieder programma gebruikt kan worden (bijvoorbeeld een Datum-object), moet er gebruik gemaakt worden van information hiding en encapsulation (data abstractie).

Leg uit wat er precies onder information hiding en encapsulation verstaan wordt en hoe ze gerealiseerd worden.
Gebruik in je uitleg de begrippen interface, class, private velden, public velden, specificatie object, implementatie object, pre-conditie en post-conditie.

- b) Maak hierna beschreven drie classes met de namen P, Q en R.

De class P is de superclass van de class Q en de class R is een subclass van de class Q.

De twee classes Q en R bevatten ieder 1 variabele van het type int met respectievelijk de namen v1 en v2;

De classes Q en R bevatten precies 1 constructor, namelijk een constructor met 1 parameter van het type int, waarmee de waarde van de variabele van die class kan worden ingesteld.
De constructor van de classe R initialiseert de via inheritance verkregen variabele op de waarde -1 d.m.v. een aanroep van de constructor uit de superclass.

De class P bevat een abstracte methode: "int kwadraat ()".

De class Q overschrijft (overrides) de methode kwadraat() met een implementatie welke het kwadraat van de globale variabele v1 retourneert.

De class R bevat een methode: "int kwadraat (int i)" welke het kwadraat van de parameter i retourneert en een methode: "int dubbele ()" welke het dubbele van de globale variabele v2 retourneert.

- c) De vragen dit onderdeel gaan over de drie classes welke in onderdeel b) beschreven zijn.

Waarom heeft de class P geen constructor te bevatten?

Als de onderstaande declaraties gegeven zijn,

```
Output out = new Output();
P p = new P(5);
Q q = new Q(10);
```

wat is dan de uitvoer van de volgende statements?

```
out.println(p.kwadraat());
out.println(q.kwadraat());
out.println(q.kwadraat(q.verdubbel()));
out.println(q.v1);
out.println(q.v2);
```

- d) Wat is een "final class"?
Wat is een "final method"?

Opgave 2.

Gegeven zijn de onderstaande classes.

```
class Knoop {
```

```
    int data;
    Knoop left, right;
```

```
    public Knoop (int d) {
        this(d, null, null);
    }
```

```
    public Knoop (int d, Knoop l, Knoop r) {
        data = d;
        left = l;
        right = r;
    }
```

```
}
```

```
public class BinaireZoekboom {
```

```
    private Knoop wortel;
```

```
    ....
    ....
}
```

- a) Maak een parameterloze constructor voor de class BinaireZoekboom die de lege binaire zoekboom aanmaakt.

- b) Implementeer in de class BinaireZoekboom de onderstaande methode hoogte() welke de hoogte van de binaire zoekboom retourneert

```
public int hoogte () {
```

```
    ....
    ....
}
```

HINT: gebruik een recursieve hulpmethode.

- c) Implementeer voor de class BinaireZoekboom een copy constructor waarmee een deep copy van de binaire zoekboom gemaakt wordt.

```
public BinaireZoekboom (BinaireZoekboom b) {
```

```
    ....
    ....
}
```

- d) Implementeer in de class BinaireZoekboom de methode insert() waarmee een integer waarde kan worden toegevoegd aan de binaire zoekboom. Geef een exception als er geprobeerd wordt om een integerwaarde toe te voegen die al aanwezig is in de binaire zoekboom.

```
public void insert (int i) throws Exception {
    ....
    ....
}
```

Opgave 3.

a) Bespreek hash tabellen.

Geef in deze bespreking duidelijk aan welk het probleem is dat men met hash tabellen wil oplossen en hoe dat deze oplossing werkt. Gebruik in je bespreking de termen "hash function", "collision" en "collision resolution". Geef in je bespreking ook een voorbeeld van een collision resolution.

b) Gegeven is de onderstaande methode merge()

```
int[] merge (int[] a, int[] b);
/* PRE - Het aantal elementen in de arrays a en b is respectievelijk
gelijk aan a.length en b.length
de elementen in a en b zijn monotoon niet-dalend gesorteerd
POST - een int-array met de samenvoeging van alle elementen van a en
b, in monotoon niet-dalend volgorde, is geretourneerd
*/
```

B.v.: Als a=(3,5,7) en b=(1,2,3,4,5) geldt: merge(a,b)=(1,2,3,4,5,5,7)

Gebruik de gegeven methode merge() om m.b.v. het mergesort algoritme de onderstaande methode mergesort() recursief te implementeren.

```
int[] mergesort (int[] a);
/* PRE - het aantal elementen in het array a is gelijk aan a.length
POST - de elementen van het array a zijn in monotoon niet-dalende
volgorde geretourneerd.
*/
```

c) Bespreek wat er allemaal aan de methode mergesort() uit onderdeel b) van deze vraag veranderd en/of toegevoegd moet worden om deze methode te kunnen laten werken voor objecten waarop een ordening gedefinieerd is.

d) Aan een lege AVL-boom waarvan de knopen integers bevatten, worden achtereenvolgens, in de gegeven volgorde, toegevoegd de getallen

25 15 35 18 10 20 12 5 1 14

Teken de drie AVL-bomen die de toestand laten zien van respectievelijk na het toevoegen van de 5, de 1 en de 14.

Beoordelingstabel

opgave	a	b	c	d	e	f	totaal
1	5	10	10	5			30
2	5	5	10	10			30
3	10	10	5	5			30
TOTAAL							90

Het eindcijfer E wordt als volgt uit het totaal T verkregen : $E = T/10 + 1$.