
Lösningförslag till tentamen *

Kursnamn	Objektorienterad programmering
Tentamensdatum	2008-03-28
Program	DAI 2
Läsår	2007/2008, lp II
Examinator	Uno Holmer

* se även www.chl.chalmers.se/~holmer/ där finns det mesta av kursmaterialet.

Uppgift 1 (10 p)

Ingen lösning ges. Se kurslitteraturen.

Uppgift 2 (8 p)

```
public class MyArrayList extends ArrayList<Integer> {
    public MyArrayList reverse() {
        Integer temp;
        for ( int i = 0, j = size() - 1; i < j; i++, j-- ) {
            temp = get(i);
            set(i, get(j));
            set(j, temp);
        }
        return this;
    }
}
```

Uppgift 3 (12 p)

```
public class Båt {
    private String namn;
    private int längd;

    public void skrivInfo() {
        System.out.println( "Namn: " + namn + "\nLängd: " + längd );
    }
}

public class Motorbåt extends Båt {
    private int motorstyrka;
    private int maxfart;

    public void skrivInfo() {
        super.skrivInfo();
        System.out.println( "Motorstyrka: " + motorstyrka +
            "\nMaxfart: " + maxfart );
    }
}

public class Segelbåt extends Båt {
    private int segelyta;

    public void skrivInfo() {
        super.skrivInfo();
        System.out.println( "Segelyta: " + segelyta );
    }
}
```

```
}  
  
public class Båtregister {  
    private ArrayList register = new ArrayList();  
  
    public void addera( Båt b ) {  
        register.add( b );  
    }  
  
    public void skrivInfo() {  
        Iterator it = register.iterator();  
        while( it.hasNext() )  
            ((Båt)(it.next())).skrivInfo();  
    }  
}
```

Uppgift 4 (17 p)

a) (7 p)

A.f
C.f

b) (9 p)

```
Base obj1 = new Sub1();  
obj1.f();  
obj1.g();  
obj1.h(); // h är odefinierad för Base
```

```
Base obj2 = new Sub2();  
obj2.f();  
obj2.g();  
obj2.h(); // h är odefinierad för Base
```

```
Int obj3 = new Sub2();  
obj3.f();  
obj3.g(); // g odefinierad för Int  
obj3.h();
```

Uppgift 5 (14 p)

```
public TreeMap<String, TreeMap<String, Integer>> sammanställPrestationer() {
    TreeMap<String, TreeMap<String, Integer>> ladok =
        new TreeMap<String, TreeMap<String, Integer>>();

    for (Studentprestation p : prestationer) {
        TreeMap<String, Integer> kursprestationer;

        if ( ! ladok.containsKey(p.getKurskod()) ) {
            kursprestationer = new TreeMap<String, Integer>();
            ladok.put(p.getKurskod(), kursprestationer);
        } else
            kursprestationer = ladok.get(p.getKurskod());

        if ( ! kursprestationer.containsKey( p.getPnr() ) )
            // Studenten har tagit sina första poäng.
            kursprestationer.put(p.getPnr(), p.getPoäng());
        else {
            // Studenten har poäng från tidigare moment.
            Integer poäng = kursprestationer.get(p.getPnr());
            kursprestationer.put(p.getPnr(), poäng + p.getPoäng());
        }
    }
    return ladok;
}
```