

# Tentamen

DD2385 Programutvecklingsteknik vt 2009

Fredagen den 5 juni 2009 kl 10.00 – 13.00

Inga hjälpmedel utom penna, sudd och linjal

Tentans del I omfattar 22 poäng. Del II har också 22 poäng

Preliminära betygsgränser (gränserna blir inte högre):

Betyg FX: 15-16 poäng på del I  
Betyg E: 17-22 poäng på del I  
Betyg C: 17-22 poäng på del I och  
11-22 poäng på del II

Betyg FX får kompletteras till C om C-nivån på del II är nådd på tentan.

Det är tillåtet att skriva lösningar till flera uppgifter på samma blad.

## del I

1. (5p) Rita ett UML-klassdiagram som åskådliggör följande klasser och interface. Alla relationer ska vara med. För full poäng måste öppna och slutna pilspetsar samt streckade och heldragna linjer användas rätt. Instansvariabler och metoder behöver *inte* vara med i diagrammet men ev. relationer som ges av variabler och metoder ska ritas ut. Alla klasser och interface som nämns måste vara med i diagrammet. Multipliciteter som inte är 1 ska skrivas ut.

Alla klasserna är samlade på nästa sida.

```

interface Model {
    public void setStep(int step);
    public void toggle(int i, int j);
    public void update();
    public int getVal(int i, int j);
}

class MiniModel implements Model {
    private int step = 1;

    public void setStep(int step) {
        this.step=step;
    }

    public void toggle(int i, int j) {
        System.out.println("This method will toggle positions");
    }

    public void update() {
        System.out.println("This method will update the model using a " +
            "complicated algorithm");
    }

    public int getVal(int i, int j) {
        return 17;
    }
}

class Demo extends JFrame {
    Display display;
    JButton[] control = new JButton[4];
    TextField input = new TextField();
    Model model;
}

class Display extends JPanel implements MouseListener {
    // all methods in mouset listener must be implemented
}

```

**2.** (1p) Vad kallas det mönster som Model och Minimodel illustrerar?  
(Det är inte MVC)

**3.** (4p)

Vad är en *inre klass* i Java?

Näm en fördel med att använda inre klasser eller beskriv kort en typisk situation där det passar att använda en inre klass.

Näm en nackdel med att använda inre klasser.

Hur ritas man i UML-klassdiagram att klassen A är en inre klass i klassen B ?

4. (3p) Nedan följer fem korta beskrivningar av designmönster. Välj ut vilken beskrivning som stämmer för var och en av

**Template Method    Proxy    Facade**

Tre beskrivningar väljs ut, två blir över.

- A) Skjut upp delar av en algoritm till subklasser.
- B) Ett objekt kontrollerar åtkomsten till ett annat objekt. De båda implementerar samma gränssnitt.
- C) Ett objekt har hand om kommunikationen mellan  $n$  st objekt.
- D) Lägg till funktionalitet till ett objekt dynamiskt. Javas klasser för strömmar följer mönstret.
- E) En enda klass ger ett gränssnitt till ett komplext system.

5.

a) (1p) Vad är namnet på det designmönster där ett objekt underrättar  $n$  st andra objekt när det första objektets tillstånd ändras?

b) (1p) Förklara kort hur mönstret tillämpas på Javas grafiska komponenter och lyssnare, dvs tala om vilken roll i mönstret en grafisk komponent respektive en lyssnare har.

6. (2p) Para ihop arbetssätten *Vattenfall* och *eXtreme Programming* med följande karakteristika: Var och en av de fyra ska knytas till något arbetssätt.

- A) Metodik lämplig för stora projekt med oerfarna programmerare
- B) Testning av systemet görs kontinuerligt (eller åtminstone ofta)
- C) Alla i projektet har insyn i och tillgång till all kod
- D) Systemspecifikation görs tidigt

7. (2p) Vad är ett "Data Dictionary" när det används som hjälpmedel inom objektorienterad analys? Hur skapas ett "Data Dictionary"?

8. Antag att klassen `Punkt` finns och har en konstruktor som stämmer med hur objekt skapas i koden nedan. Antag också att satserna i koden exekveras i en `main`-metod.

a) (1p) Hur många objekt av klassen `Punkt` har skapats totalt vid  
`// *** a) *** ?`

b) (1p) Vilka två tal skrivs ut av utskriftssatserna på slutet ? Det är bara talens värden och ordning som efterfrågas, inte hur utskriften ser ut för övrigt. Båda talen måste vara rätt för 1p.

```
ArrayList<Punkt> pm1 = new ArrayList<Punkt>();

pm1.add(new Punkt(0, 7, 13));
pm1.add(new Punkt(-1, 5, 8));
pm1.add(new Punkt(3, 6, -2));

ArrayList<Punkt> pm2 = new ArrayList<Punkt>();
for (Punkt p : pm1)
    pm2.add(p);

    // *** a) ***

pm1.remove(1);
System.out.print(pm1.size() + " ");
System.out.println(pm2.size());
```

b) (1p) Om man vill sortera en lista av punkter kan man t.ex. använda sig av metoden `sort`

```
Collections.sort(punktlista);
```

där `punktlista` har typen `ArrayList<Punkt>`. Vad måste läggas till i klassen `Punkt` för att sorteringen ska gå att utföra ?

## del II

9. (2p) Vilket av följande påståenden stämmer för en instansmetod i klassen A vars deklaration inleds med `synchronized void syncm()` ?

- A) Under exekvering av `a.syncm()` kan inga andra metoder i klassen A märkta `synchronized` anropas.
- B) Under exekvering av `a.syncm()` sker en synkroniserad skräpsamling.
- C) Under exekvering av `a.syncm()` är objektet som `a` refererar till låst så att andra trådar inte kan anropa metoder märkta `synchronized` i objektet `a`.
- D) Under exekvering av `a.syncm()` är objektet som `a` refererar till låst så att andra trådar inte kan anropa metoden `syncm()` i `a`.

10. (2p) `static` är svårt. Ibland måste man använda det och ibland får man inte. Vad gäller här ? Markera `static` eller `icke-static` för den variabel eller metod som beskrivs i påståendena.

- A) En metod anropas `Collections.sort(lista)`, dvs metoden anropas via ett klassnamn. Är metoden `sort()` deklarerad `static` ?
- B) Den variabel som inuti en Singleton - klass refererar till Singleton-objektet, är den deklarerad `static` ?
- C) Metoden `actionPerformed(...)` som skrivs när man implementerar ett lyssnargränssnitt, skall den deklareraras `static` ?
- D) Om man inom klassen Punkt vill räkna hur många Punkt-objekt som skapas, Ska `antalP` vara `static`?

```
class Punkt {
    int antalP; // static eller ej ???
    int x,y,z;
    Punkt(int x, int y, int z) { ... antalP++; ... }
}
```

11. (3p)

Skriv xml som beskriver uppgifterna på tentans del II med rubrik, nummer (inklusive a,b,...) och poäng. Du kan använda rubrikerna `sync`, `static`, `xml` respektive femtonspel. Ingen speciell DTD behöver följas men koden ska vara välformad och entydig.

**12.** (15p)

Ett Femtonspel som spelas på en datorskärm består av  $4 \times 4$  (alltså 16) klickbara rutor. En ruta är tom och övriga innehåller talen 1, 2, 3, ... 15. Detta är spelets basversion. Talen kan bytas ut mot bilder. Det finns en eller flera naturliga ordningar mellan rutorna. När spelet startar är talen (eller motiven) slumpmässigt utspridda och spelet går ut på att bringa rutorna i ordning genom att klicka på rutor som gränsar med en kant till den tomma rutan. Det klickade motivet flyttas då till tomma rutan och den klickade rutan blir tom. Klick i rutor som inte gränsar till den tomma har ingen effekt eller möjligen ges ett vänligt felmeddelande. Om man vill mäta en spelprestation för att kunna spela mot sig själv eller någon annan kan man t.ex. räkna hur många klick som behövs och/eller hur lång tid spelet tar.

**a)** (1p)

Det är klokast att lägga spelets logik (modell) i en egen klass men bör man använda M-V-C, dvs dela upp programmet i de tre delarna Model, View och Control eller är det bättre att låta Model vara för sig och View och Control tillsammans, alltså en uppdelning M-VC ? Motivera svaret!

**b)** (2p)

När spelet startar ska talen 1,2, ... 15 i slumpmässig ordning visas upp på spelplanen. Skriv de satser som behövs för att åstadkomma en `ArrayList<T>` eller annan `Collection`-klass som innehåller talen i slumpmässig ordning. Du ska ange lämpligt "värde" på T. Till hjälp finns en metod som kan anropas `Collections.shuffle(tallista)` som blandar listan tallista slumpmässigt. (Vi struntar i att det (troligen) krävs en viss typ av permutation för att spelet ska gå att lösa.)

**c)** (2p) I Femtonspelets implementation är det viktigt att kunna svara på frågan om en klickad ruta ligger bredvid den tomma rutan. Antag att rutorna har koordinater och att tomma rutan har (it, jt) och den klickade rutan har (ik, jk). Skriv ett villkor som är sant om de ligger bredvid varandra. Det förutsätts att klickad ruta är inom spelplanen. Det behöver inte vara riktig Java-syntax, ett matematiskt villkor duger bra.

**d)** (5p)

Skissa modellklassen genom att svara på/lösa följande:

Vilken datastruktur är lämplig för att representera spelplanen i modellklassen?

Skriv huvuden, dvs typ, namn och parameterlista för de metoder som modellklassen ska ha. Beskriv i ord vad metoderna ska göra. Du behöver inte skriva Javakod annat än i metodhuvudena. Ange vilka metoder som används för att kommunicera med View och/eller Control och vilka som bara används inom modellen.

**e)** (4p)

Tänk igenom vilka klasser och interface som grafik- och interaktionsdelen av programmet ska bestå av och rita ett UML-klassdiagram över detta. Fundera på följande: Ska man använda klasser från biblioteken som de är eller ska det skrivas några subklasser? Rita gärna upp en bild av hur det grafiska gränssnittet ska se ut. Det ger inga poäng men kan vara till hjälp när klassdiagrammet ska konstrueras.

**f)** (1p)

Antag att Femtonspelstävlingar ordnas på så sätt att många spelare ansluter sig till en server. Servern ger alla spelare samma startställning och startar tidtagning för var och en. När en spelare är klar så skickar hon/han lösningen till servern som avgör om spelet är slut och i så fall meddelar alla spelare vem som vann. Vilket designmönster tillämpas då?