

## Innlevering 2 PG2100

Frist søndag 11. mars 2012 kl. 23:55

- **Oppgaven skal løses individuelt.**
- Oppgaven (kildekoden og eventuelt andre dokumenter som er relevante for løsningen) leveres på ItsLearning som *én* .zip-fil med navn på formen etternavn\_fornavn, f.eks. **einstein\_albert.zip**.
- **Alle deler av innleveringen skal ha navnet på den som har løst oppgaven som en kommentarlinje i starten.**
- **Det legges generelt vekt på:**
  - Ryddig kode (luft, innrykk), standard navngiving av klasser, metoder og variabler, fornuftige datatyper, klargjørende kommentering av kode, kjørbart program, programmet virker som tiltenkt.
  - Oversiktlige beskrivelser og forklaringer.
- Det er greit hvis du vil bruke engelsk ved navngiving av variabler osv.
- *Oppgi ellers eventuelle forutsetninger du gjør.*

Pass på at innleveringen oppfyller kravene over!

## Oppgave 1

Klassene C og D er gitt i rammene under.

```
public class C {
    private int m = - 1;
    private int n = - 2;

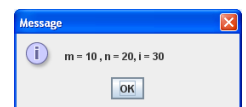
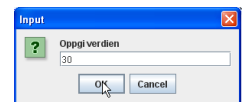
    public C () {
        m = 1;
        n = 2;
    }
    public C (int m, int n) {
        setM(m);
        setN(n);
    }
    public int getM() {
        return m;
    }
    public int getN() {
        return n;
    }
    public void setM(int m) {
        this.m = m;
    }
    public void setN(int n) {
        this.n = n;
    }
    public void metodel() {
        System.out.println(
            "C sin metodel!");
    }
    public String toString() {
        return "m = " + m +
            ", n = " + n;
    }
}
```

```
public class D extends C {
    private int i = -3;

    public D(int i) {
        setI(i);
    }
    public int getI() {
        return i;
    }
    public void setI(int i) {
        this.i = i;
    }
    public void metodel() {
        System.out.println(
            "D sin metodel!");
    }
    public void metode2() {
        metodel();
    }
    public String toString() {
        return "i = " + i;
    }
}
```

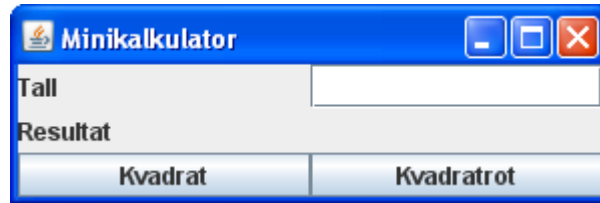
```
public class Klientprogram {
    public static void main(String[] args) {
        int tall = 3;
        D d = new D(tall);
        d.metode2(); //x
    }
}
```

- Hvilke verdier har attributtene i, m og n etter at objektet er opprettet i main-metoden når Klientprogram kjøres? Forklar hvorfor de har disse verdiene.
- Endre klientprogrammet slik at verdien for variabelen tall blir gitt ved bruk av en JOptionPane sin input dialogboks.
- Legg til kode i klientprogrammet som sørger for at attributtenes verdier blir vist ved bruk av JOptionPane sin message dialogboks og på samme format som toString-metodene bruker.
- Hva viser utskriften når setningen merket //x blir utført? Forklar hvorfor utskriften blir som den blir.
- Skriv en ny konstruktør i klassen D som har parametre for alle de tre attributtene i, m og n, og som sørger for at attributtene får disse parameterverdiene. Vis to måter dette kan gjøres på.
- Endre toString-metoden i klassen D slik at returverdien viser hva verdien til alle de tre attributtene er. Vis to måter dette kan gjøres på.
- Legg til kode i klientprogrammet som sørger for at attributtenes verdier blir vist ved bruk av objektets toString-metode og JOptionPane sin message dialogboks (som i c)).



## Oppgave 2

Skriv en applikasjon som setter opp følgende grafiske brukergrensesnitt (GUI):



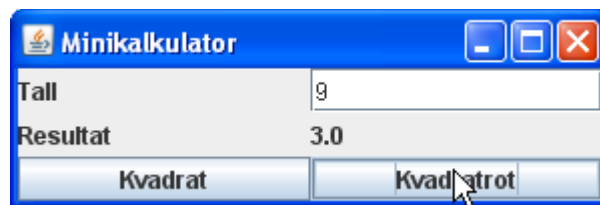
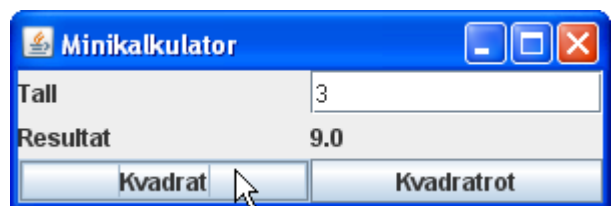
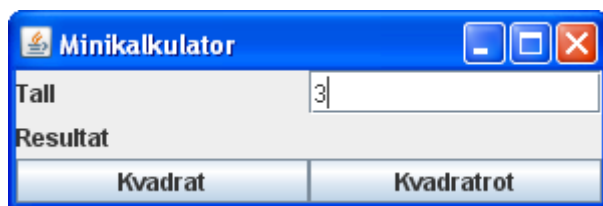
Vinduet inneholder etiketten "Tall" (JLabel) og et tekstfelt (JTextField), to knapper "Kvadrat" og "Kvadratrot" (JButton) og etiketten "Resultat" og en blank etikett.

Størrelsen på vinduet er 300 x 100, og layout i vinduet er GridLayout.

Ved klikk på knappen "Kvadrat", blir tallet i tekstfeltet kvadrert (ganget med seg selv), og resultatet vises i den blanke etiketten.

Ved klikk på knappen "Kvadratrot", blir kvadratroten av tallet beregnet og vist som over.

Se eksempler på kjøring under:



Kvadratrot av et tall kan beregnes ved bruk av `Math.sqrt(tall)` (sqrt – square root).

Applikasjonen skal avsluttes ("exit") ved lukking av vinduet (se forelesning 9).

----- Slutt på oppgaven -----

Evalueringskriterium

(rad 3 viser vekten for hver deloppgave)

Oppgave 1							Oppgave 2										Helhet			
							Komponenter (deklarasjon, oppretting og utlegging)			Layout	Lytter			Oppsett				main-metoden	Kodestandard	Ryddighet
							JLabel	JTextField	JButton	GridLayout(3, 2)	Implementere ActionListener	actionPerformed	addActionListeners	dimensjonering	synliggjøring	defaultCloseOperation				
a	b	c	d	e	f	g	3 stk	1 stk	2 stk											
3	5	5	3	5	5	5	4	4	4	3	4	4	3	2	2	2	3	4	4	