

Examination in  
**PROGRAMMERINGSTEKNIK F1**

TIN211

DAY: TUESDAY    DATE: 2008-12-16    TIME: 8.30-12.30

ROOM: H

Responsible teacher:    Erland Holmström tel. 1007, home 270358

Results:    Are sent by mail from Ladok.

Solutions:    Are eventually posted on homepage.

Inspection of grading:    The exam can be found in our study expedition after posting of result.

Time for complaints about grading are announced on homepage after the result are published or mail me and we find a time.

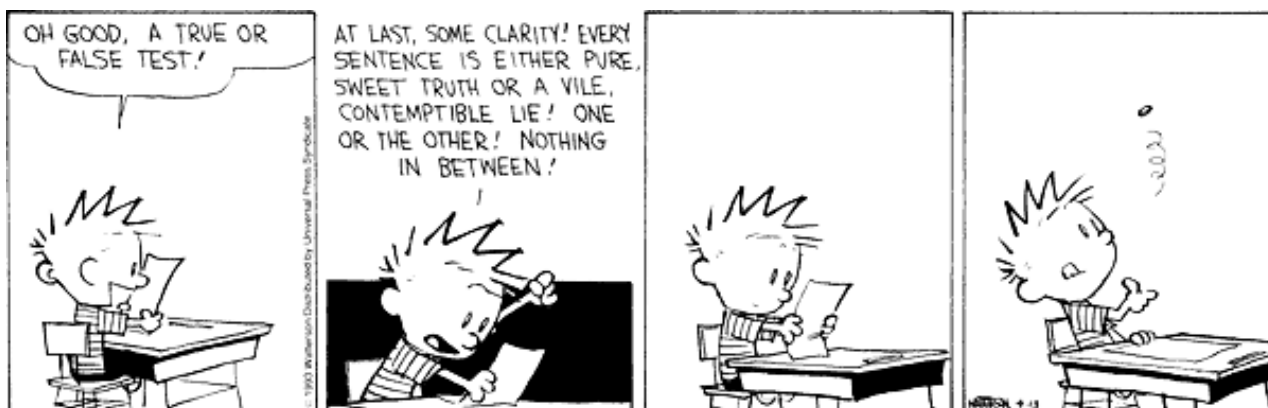
Grade limits:    CTH: 3=26p, 4=36p, 5= 46p, max 60p

Aids on the exam:    **Horstmann Java Concepts both 4<sup>th</sup> and 5<sup>th</sup> edition are ok,  
and A print of Chapter 18.**

or Skansholm Java direkt if you took the course with Bondesson.

### Observe:

- Start by reading through all questions so you can ask questions when I come. I usually will come after ca 2 hours.
- All answers must be motivated when applicable.
- Write legible! Draw figures. Solutions that are difficult to read are not evaluated!
- Answer concisely and to the point.
- The advice and directions given during course must be followed.
- Programs should be written in Java, indent properly, use comments and so on.
- Start every new problem on a new sheet of paper.

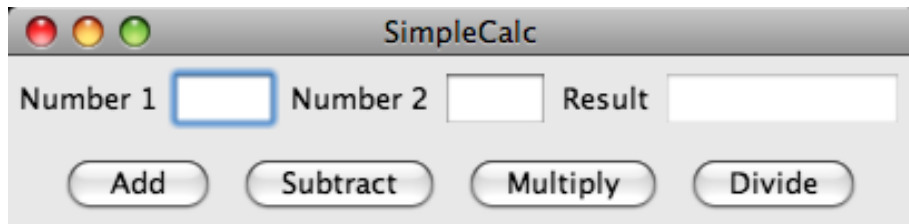


**Good luck!**

Problem 1. Motivera...

- Är " $5 < a \ \&\& \ 3 > b$ " detsamma som " $!(5 < a) \ || \ !(3 > b)$ "?
- Vad är värdet av  $1653/100$  och värdet av  $1653\%100$  :-)
- Antag `int x=12`; Om `y` är en statisk metod, kan man då göra `x.y()`?  
Går det om `y` är en instansmetod? (5p)

Problem 2. Skriv ett program som fungerar som en enkel kalkylator enligt figuren nedan



När man skriver in tal i de 2 första textfälten och klickar på "Add", "Subtract", "Multiply" eller "Divide" så utförs motsvarande operation och resultatet skrivs i Result fältet. Om man inte skriver in något tal eller ett felaktigt tal i "Number 1" eller "Number 2" så skall en exception kastas och fångas. Programmet skall ha en metod enligt:

`void calculate(char operator)` som läser talen från textfälten, kastar ev. en exception (men du fångar den där du anropar `calculate`), utför operationen "operator" och skriver resultatet.

(21p)

Problem 3. Inom geometrin finns det mycket hierarki. Tex är varje sida i en polygon en linje, varje linje består av två punkter och en punkt består av två reella tal (eller heltal, det är inte så noga).

**a)** En polygon kan representeras med en `ArrayList` med punkter. Skriv en klass `Polygon` som förutom en konstruktor har metoderna

`void add(Point p)` - som lägger till en punkt till polygonen (`Point` är ungefär samma sak som `Point2D.double` men kortare att skriva)

`double perimeter()` - som beräknar dess omkrets (= summan av längderna på linjerna som sammanbinder punkterna).

Polygonen antas vara slutna och punkterna har matats in medsols.

**b)** Skriv en `main` klass med en metod, `readPolygon`, som läser in en polygon (dvs punkterna) från användaren och lägger in i klassen i a). Du kan anta att tal matas in korrekt.

`Main` skapar en `polygon`, läser in punkter från användaren och beräknar omkretsen.

Avståndet mellan 2 punkter `p1` och `p2` beräknas med formeln:

$$|p1, p2| = \sqrt{(p2_x - p1_x)^2 + (p2_y - p1_y)^2}$$

(10p)

Problem 4. I Javas API hittar vi: (det mesta av Date är “deprecated” dvs skall inte användas men det spelar ingen roll för vårt exempel) “The class Date represents a specific instant in time, with millisecond precision. Specifically, a year  $y$  is represented by the integer  $y - 1900$ .” En av Dates konstruktörer och några av dess metoder är:

**Date()** Allocates a Date object and initializes it so that it represents the time at which it was allocated, measured to the nearest millisecond.

**int getYear()** Returns a value that is the result of subtracting 1900 from the year that contains or begins with the instant in time represented by this Date object, as interpreted in the local time zone.

**void setYear(int year)** Sets the year of this Date object to be the specified value plus 1900. ...“

Antag nu att vi skriver följande program:

```
public final class Period {
    private final Date start;
    private final Date end;
    /**
     * @param start the beginning of the period
     * @param end the end of the period
     */
    public Period(Date start, Date end) {
        if (start.compareTo(end) > 0) {
            ... kasta en exception
        } else {
            this.start = start;
            this.end = end;
        }
    }
    public Date start() {return start;}
    public Date end() {return end;}
    public int getEndYear() {return end.getYear();}
    // ...possibly more of no concern

    public static void main(String[] args) {
        Date start = new Date();
        Date end = new Date();
        Period p = new Period(start, end);
        System.out.println("Year= " + p.getEndYear());
        end.setYear(78);
        System.out.println("Year= " + p.getEndYear());
        p.end().setYear(79);
        System.out.println("Year= " + p.getEndYear());
    }
}
```

Den första println satsen skriver ut 108. Vad skriver dom 2 andra println satserna ut? Du måste förklara varför dom skriver det dom skriver dvs vad som sker vid anropen, rita gärna figurer. (Observera att start och end är “private final” i klassen)

Problem 5. Att fylla ett slutet område. Bilden till höger är en sk. rasterbild som schematiskt kan beskrivas som ett rutmönster i vilket en ruta (pixel) kan vara tänd eller släckt. De fyllda kvadraterna markerar svarta punkter (i verkligheten är de cirklar), alla övriga punkter är vita. (Vita punkter syns ej.)



Skriv en rekursiv metod, som givet en punkt  $(x, y)$  innanför den svarta randen fyller hela området innanför randen med svarta pixlar.

```
public void fill(char[][] m, int x, int y)
```

Det kan förutsättas att det omslutna området är sådant att man från en godtycklig punkt inom det kan komma till varje annan genom att förflytta sig vertikalt och horisontellt (men ej diagonalt). Det kan också förutsättas att området verkligen är slutet så det går inte att "smita ut" om man bara förflyttar sig en pixel vertikalt eller horisontellt.

Parametern  $m$  är en representation av bilden i form en matris med ettor (svarta pixlar) och nollor (vita pixlar)

Exempel på en matris,  $m$ , hittar du till vänster nedan. (Observera att matrisens utseende inte stämmer riktigt med figuren ovan).

Exempel på utdata efter anrop av `fill(m, 2,4)`; till höger.

000111000000	000111000000
001000100000	001111100000
001000100000	001111100000
001000111100	001111111100
010000001000	011111111000
010000010000	011111110000
001000100000	001111100000
000111000000	000111000000

Tips. Du skall INTE testa om dina grannar är tända/släckta. Det måste dom göra själva. Du ansvarar bara för dig själv.

Du kan förutsätta att matrisen existerar och att det inte finns tomma rader.

(6p)

Problem 6. Antag att du skapat klasser för olika geometriska objekt som punkter, cirklar, kvadrater, polygoner osv. Skriv en polymorf utskriftsrutin som givet ett fält med geometriska objekt skriver ut (en representation av) objekten. Vilka förutsättningar måste vara uppfyllda i de olika geometriska klasserna för att det skall fungera?

(5p)

Problem 7. I ett datorsystem för en skola har vi (java)klasser för studenter, lärare, assistenter, anställda, sekreterare, rektor, vaktmästare, personer, kurser, seminarier, lektioner, datorlab, lokaler, skola, föreläsningssalar. Beskriv vilka relationer som är lämpliga mellan dessa. Använd UML notation men beskriv pilarnas betydelse.

(5p)