

Kalle Prorok,
Thomas Johansson

Tentamen i
Objektorienterad programmering för ingenjörer
(TDBB09)
1999-05-31, kl 09.00 - 15.00

Lokal: Skrivsal 1, Östra Paviljongerna, Ålidbacken 23

Hjälpmedel: inga (förutom penna, suddgummi och matsäck)

Kursutvärdering: Sist bland papperen finns en kursutvärderingsblankett. Ta med den hem och fyll i den. Lämna in den till Thomas när du hämtar ut den rättade tentan.

Börja varje uppgift på ett nytt blad.

Använd bara en sida av papperet.

Skriv namn på varje blad.

Lägg uppgifterna i rätt ordning.

Observera att uppgifterna inte är numrerade efter svårighetsgrad och kom ihåg att även delvis lösta uppgifter kan ge poäng.

Kommentera noga all källkod så att vi kan se hur du har tänkt.

Maxpoäng är 40, fördelade på 8 frågor.

Betygsgränser:

5	32p
4	26p
3	20p

Fråga om det är något som är oklart ! Vi kommer till skrivsalen ca **10.30** och **12.00**.

Lycka till !

Thomas Johansson
tel 786 62 59

Kalle Prorok
tel 786 50 18

Uppgift 1 (3p)

- a) Beskriv hur STL är organiserat och hur en iterator fungerar. (2p)
- b) Ange minst 2 samlingstyper (1p)

Uppgift 2 (6p)

Några C++ frågor:

- a) Hur fungerar `dynamic_cast<>()` ? (2p)
- b) Varför måste man skriva `delete []` (med klamrar) när man allokerat en array med element och vilken språkfilosofi ligger bakom? (2p)
- c) Hur (i princip) hanterar språket C++ funktionsnamn som är lika men har olika parametertyper/antal? Vad kallas egenskapen (flera möjliga)? (2p)

Uppgift 3 (3p)

Några designfrågor:

- a) Vad ska man ha en abstrakt basclass till och hur gör man en? (2p)
- b) Vad är MVC-modellen? (1p)

Uppgift 4 (3p)

Beskriv tre skillnader mellan Java och C++.

Uppgift 5 (6p)

Det finns (minst) fyra sätt i C++ att ordna så att metoder i en klass kan komma åt attribut i en annan klass.

- a) Beskriv alla fyra sätt. (4p)
- b) En av metoderna är oftast den ur OO-synpunkt mest fördelaktiga, vilken, och varför? (1p)
- c) För de övriga tre, ange varför de är mindre bra. (1p)

Uppgift 6 (8p)

Banverket behöver din hjälp med en OO-design av Botniabanan. Gör ett klassdiagram med lämpliga klasser såsom tågtyper, växlar, räls, stationer, signaler och annat som du tror kan ha betydelse. Skissa på ett scenario "programsnutt"/"användningsfall" som kör ett godståg från Ö-vik till Umeå.

Kom ihåg att förklara vilken notation du använder för diagrammet (eget, Booch, OMT eller UML) och förklara också vad symbolerna betyder. Diagrammet måste innehålla minst ett vardera av aggregat, association, arv och abstrakt basclass. Det måste också finnas minst åtta olika klasser.

Uppgift 7 (6p)

Implementera en templateklass med vektorer som kan hantera godtyckliga element och som hanterar indexeringsfel via undantagshantering. Du behöver ej implementera/använda dynamiska storlekar.

Följande program skall vara körbart (och orsaka texten Indexfel!):

```
#include "myvect.h"

void main(void)
{
    v<double> a(10); // 10 element a[0] .. a[9]
    a[5] = 12.3;
    try
    {
        if (a[789] < 45.6)
            cout << a[789];
    }
    catch(RangeErr)
    {
        cerr << "Indexfel!";
    }
}
```

Uppgift 8 (5p)

Implementera en klass Tid som skriver ut objekten i formen HH:MM:SS men lagrar tiden internt i sekunder. Man ska även kunna skapa och addera Tid-objekt med + operatorn.

< SLUT PÅ UPPGIFTER >