

## E K S A M E N

**Emnekode:** IS-202  
**Emnenavn:** Emner i programmering

**Dato:** 22. februar 2005  
**Varighet:** 0900 - 1300

**Antall sider inkl. forside:** 4

**Målform:** Bokmål

**Tillatte hjelpemidler:** Ingen

**Merknader:** Les gjennom alle oppgavene og tenk gjennom hvordan du vil løse dem før du begynner å skrive.

Oppgavene kan stort sett løses uavhengig av hverandre, men informasjon som er gitt under en oppgave kan være relevant for løsning av de andre.

---

## Oppgave 1 – Datastruktur (teller 30%)

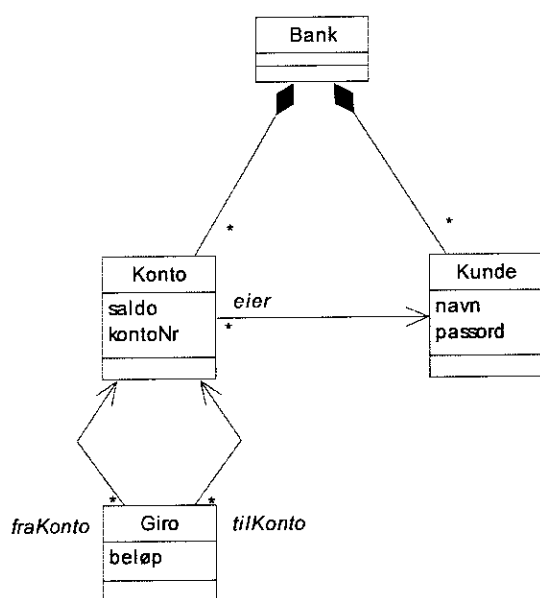
Denne oppgaven tar for seg serversiden av et (sterkt forenklet) nettbanksystem. Hver bankkonto er representert ved et objekt av typen Konto, med attributtene kontonummer og saldo. Kontoen har også en eier, som er kunde i banken. Kundene er representert ved objekter av typen Kunde, med attributtene navn og passord. En kunde kan eie mer enn en konto.

Det skal selvfølgelig være mulig å bruke nettbanken til å betale regninger.

Betalingstransaksjoner er representert ved objekter av typen Giro, som har attributtene beløp, frakonto og tilkonto.

Selve banken er representert ved et objekt av typen Bank. Bank-objektet "inneholder" alle kunde- og konto-objektene. Vi antar at alle objektene ligger i minnet til datamaskinen (i et virkelig system ville de vært lagret i en database). Dataene må lagres slik at behandlingen av betalingstransaksjoner går raskt. Tiden det tar å sette inn og fjerne objekter er ikke viktig. Antall konti er ganske stort så vi bør ikke bruke for mye minne.

Et klassediagram for systemet kan se slik ut:



- Hva slags datastruktur bør brukes for å lagre Kunde- og Konto-objekter i klassen Bank? Begrunn svaret. Finnes det klasser i standardbiblioteket til java som kan brukes?
- Når vi skal lage kontoutskrifter kan det være hensiktsmessig å ha en datastruktur Konto som inneholder giroer til og fra kontoen. Hva slags struktur bør vi bruke her, og hvorfor?
- Skriv klassedeklarasjoner for klassene Konto, Kunde, Giro og Bank. Du kan legge til flere attributter hvis du trenger dem for å løse de andre oppgavene.

## Oppgave 2 – Girering (teller 40%)

Når brukeren av nettbanken har fylt ut en giro og trykker "Betal"-knappen, kan vi anta at serveren kaller metoden girer i bank-objektet, sender med det brukeren har fylt ut som parametere. (Det kan for eksempel finnes en servlet som gjør dette. Du skal ikke skrive denne koden)

Metoden girer kontrollerer først at giro.sender er navnet på eieren til giro.frakonto, at giro.mottager er navnet på eieren til giro.tilkonto, og at password er likt passwordet til eieren til giro.frakonto. Hvis ikke dette stemmer kaster metoden en IkkeAdgangException.

Hvis alt stemmer sjekker metoden at det er dekning på frakontoen. Hvis det ikke er dekning kastes en IkkeDekningException. Hvis det er dekning lager den et nytt Giro objekt for transaksjonen, og legger det nye Giro-objektet inn i datastrukturen. Til slutt trekkes beløpet fra saldoen på frakontoen, og legges til saldoen på tilkontoen.

Skriv ferdig metoden girer i klassen Bank (og eventuelle hjelpemetoder som du trenger):

```
public boolean girer(String sender, String password, int fraKontoNr,
                    int beløp, int tilKontoNr, String mottager)
    throws IkkeAdgangException, IkkeDekningException {
}
```

Klassene IkkeAdgangException og IkkeDekningException er subklasser av Exception. Du trenger ikke skrive deklarasjoner for dem.

### Oppgave 3 – Kontoutskrift (teller 30 %)

Når nettbankbrukeren velger kontoutskrift vil web serveren kalle metoden lagKontoUtskrift i bankkobjektet.

Metoden lagKontoUtskrift tar et kontonummer, kundenavn og passord som parametere. Den sjekker først at navn og passord er riktig for eieren av kunden. Hvis navn og/eller passord er feil kaster metoden en IkkeAdgangException, ellers returnerer metoden kontoutskriften som en String. Kontoutskriften skal se slik ut:

```
”Kontoutskrift for Espen, kontonr: 12345
Saldo: kr 10000000000000000000
Siste 10 transaksjoner:
Til Per, konto 54321, kr 100
Fra Kongen, konto 10101, kr 10000000000000000000
Til Pål, konto 14532, kr 100
Fra Prinsessa, konto 10102, kr 10
...
(10 transaksjoner)
”
```

Skriv metoden lagKontoUtskrift i klassen Bank:

```
public String lagKontoUtskrift(int kontoNr, String kunde, String passord)
    throws IkkeAdgangException {
}
```