

Fakultet for teknologi

Eksamensoppgave i TDAT1001 Programmering grunnkurs, S2

Faglig kontakt under eksamen: Grethe Sandstrak, Bjørn Klefstad

Tlf.: 73 55 95 61/ 73 55 95 72

Eksamensdato: 3. juni 2016

Eksamenstid (fra-til): 09:00 – 13:00

Hjelpemiddelkode/Tillatte hjelpemidler: Alle trykte hjelpemidler

Annen informasjon:

Målform/språk: Bokmål

Antall sider (uten forside): 2

Antall sider vedlegg: 0

Oppgave 1

Du skal utvikle en Java-applikasjon for enkel byplanlegging for ett land. Du trenger dermed kun å forholde deg til ett land i gangen. Ett land har plass til et maks antall byer, den enkelte by har plass til et maks antall innbyggere. I oppgavene nedenfor er det beskrevet mer i detalj hva som skal løses. Krav til koden: Alle klasser skal lages etter prinsippet *komposisjon*. Metoder skal ha fornuftige returverdier og feilsituasjoner skal håndteres. Dersom du trenger flere metoder for å løse oppgavene lag også disse – husk da å angi hvilke klasser disse tilhører.

- a) Les gjennom hele oppgaveteksten og lag utvidet klassediagram for de klassene du trenger for å løse oppgavene. Husk å få fram sammenhengen mellom klassene.
- b) Lag klassen *By*, den enkelte by kjennetegnes ved at den har et navn, antall innbyggere, maks antall innbyggere og navnet på byens ordfører. Klassen skal ha to konstruktører - en som gir verdi til alle objektvariabler og en standardkonstruktør. Videre skal klassen ha get/set metoder og en toString-metode. Det skal ikke være mulig å opprette en by med flere innbyggere enn det er plass til. Skjer dette, skal det kastes et unntak.
- c) Lag en metode i klassen *by* som sjekker om en by er lik en annen by. To byer er like dersom de har samme navn.

Du skal nå jobbe med klassen *Land*. Et land har et navn, navn på statsoverhode og 0,1 eller flere byer. Hvert land har en øvre grense for maks antall byer det er plass til.

- d) Sett opp klassen *Lands* objektvariabler, lag en passende konstruktører og lag en toString-metode som lister ut Landets navn, navn på statsoverhodet og informasjon om alle registrerte byer.
- e) Lag en metode i klassen *land* for å registrere en ny by. Ny by kan kun registreres dersom byen har korrekt informasjon om innbyggertall, ikke er registrert fra før og dersom det er plass til byen i landet.
- f) Lag en metode i klassen *land* som returnerer antall innbyggere totalt i landet.
- g) Lag en metode i klassen *land* som returnerer alle byer som har plass til et gitt antall nye innbyggere.
- h) Lag en metode som returnerer en tabell med navnene til alle ordførere i et land. Navnene skal være sortert i alfabetisk rekkefølge.

- i) Lag en JUnit - @Test-metode som tester metoden for å registrere ny By i klassen Land. Du skal teste flere ulike ting i samme test-metode:
- Registrering av en enkelt by, med korrekte inndata
 - Registrering av en by som er registrert fra før
 - Registrering av en by som har for mange innbyggere i forhold til hva byen har plass til
 - Registrering av en by mer enn det landet har plass til.
- j) Lag to klassemetoder som henholdsvis:
- skriver et objekt av typen Land til ei fil. Dersom fila ikke eksisterer skal den opprettes, eksisterer den skal de nye dataene skrive over de gamle
 - leser et objekt av typen Land fra fil og returnerer dette.
- k) Lag et menystyrt klientprogram som lar bruker registrere nytt land, registrere ny by, liste ut alle ordførere, liste ut alle byer til et land og avslutt. Ved registrering av nytt land skal eventuelt tidligere registrert land slettes. Du trenger dermed kun å forholde deg til ett land i gangen. Dersom land ikke er registrert skal det ikke være mulig å Når bruker avslutter skal alle registrerte data om land og byer lagres på fil. Ved oppstart skal eventuelle lagrede data leses inn.