IN1000 - Seminaroppgaver til uke 11

Fortsettelse på Studentsystemet:
Du har allerede laget Fag-objekter som holder oversikt over sine Studenter, og Student-objekter som holder oversikt over sine Fag. Nå skal dere lage et objekt "StudentSystem" som skal knytte alle fag sammen og holde oversikten over alle studenter og alle fag, samt å tilby en del operasjoner for brukeren.

Programmet skal tilby
følgende operasjoner:
• Legge til ny student.
• Legge til nytt fag.
• Legge til et nytt fag til en student/legge til en ny student til et fag.
• Skrive ut alle studenter som tar et fag.
• Skrive ut alle fag en student tar.
• Finne ut hvilket fag som blir tatt av flest studenter.
• Finne ut hvilken student som tar flest fag.
• Fjerne en student fra et fag/fjerne et fag fra en student. (Legg til metoder i Student og Fag)
• Avslutt program.

Programmet skal lese inn fra en gitt fil; emnestudenter.txt. Filen er på følgende format (filinnlesningen skal løses i oppgave 1.2):

*MAT1001
Emilie
Jonathan
Henrik
Mari
Esben
Gard
Fridtjof
*MAT1100
Mathias
Ole Petter
Elsie
Emilie
Andreas
Martin

Oppgave 1
1.1 Lag en klasse "Studentsystem". Skriv konstruktøren for Studentsystem, denne skal ved opprettelsen ikke ta inn noen parametre, men skal lage en tom liste som skal holde på studenter og en tom liste som skal holde på fag.

1.2 Lag en metode i Studentsystem som håndterer innlesning fra fil, hvor filnavnet sendes med som parameter. Alle studenter skal legges inn i listen over Studenter og alle Fag inn i listen over

Fag. Pass på så du ikke legger til duplikater av studenter (anta at alle studenter har unikt navn). Her kan du ha bruk for metoden du skriver i 1.3.
Merk: studentene må i tillegg meldes opp til faget de ligger under (legges til i Fagets liste over studenter) og studentene må legge til faget i sin oversikt/liste over fag.

1.3 Lag en metode i Studentsystem som finner og **returnerer** en Student basert på navn. Metoden skal returnere None dersom studenten ikke finnes i listen. *Merk: her må du bruke student-objekenes "hentStudentNavn()-metode.*

1.4 Lag en metode i Studensystem som finner og **returnerer** et Fag basert på navn/fagkode.

Oppgave 2
2.1 Lag en metode som ber om et Student-navn (input fra bruker) og skriver ut alle fagene som den aktuelle studenten tar. Dersom studenten ikke finnes skal en beskjed skrives ut til brukeren.
Hint: Bruk metoden du skrev i 1.3 samt Student sin metode "skrivFagPaaStudent()"

2.2 Lag en metode som ber om et Fag-navn (input fra bruker) og skriver ut alle studentene som tar det aktuelle faget. Dersom faget ikke finnes skal en beskjed skrives ut til brukeren.

2.3 Lag en metode som finner ut hvilken student som tar flest fag. Resultatet kan skrives ut i terminalen. *Hint: Bruk studentenes hentAntallFag()-metode.*

2.4 Lag en metode som finner ut hvilket fag som blir tatt av flest studenter. Resultatet skrives ut i terminalen.

Oppgave 3
3.1 Lag en metode som legger til en ny student i systemet, husk å sjekke om studenten allerede finnes.

3.2 Lag en metode som legger til et nytt fag i systemet, husk å sjekke om faget allerede finnes.

Oppgave 4
4.1 Lag en metode for å registrere en student ved ett fag (hvor studentens og fagets navn gis som input fra bruker). Husk å sjekke at både studenten og faget er registert (finnes i listen over alle fag og alle studenter), samt at studenten ikke allerede tar faget.

Tips for å sjekke om studenten allerede tar faget; legg til metoden "tarFag()" i student, som tar inn et fag og returnerer True dersom den finnes i studentens liste over fag, og false dersom faget ikke finnes, feks slik:

```
    def tarFag(self, fag):
        return fag in self._fagliste
```

Oppgave 5
Lag en metode i Studentsystem som håndterer ordreløkken. For hver ordre skal det kalles på en metode. Lag gjerne en metode som heter "skrivMeny()" som kun skriver ut oversikten over alle valgene.

Oppgave 6
Lag et lite hovedprogram som oppretter et studentsystem. Les inn filen "fagliste.txt" (kall på studentsystemets innlesnings-metode). Start deretter ordreløkken.

Oppgave 7 - Hvis tid
Lag en metode som sletter en student fra ett fag (hvor studentens og fagets navn gis som input fra bruker). Husk å sjekke at både studenten og faget er registert (finnes i listen over alle fag og alle studenter), samt at studenten må ta faget.

NB! Her bør man legge til en metode i Fag og en metode i Student, disse er gitt under:

I student:

```
    def fjernFag(self, fag):
        self._studentliste.remove(fag)
```

I fag:

```
    def fjernStudent(self, student):
        self._studentliste.remove(student)
```