Oppgave 4. Reservasjonssystem (50 poeng)

I denne oppgaven skal du skrive en første versjon av et program for reservasjon av rom på ett eller flere hoteller. I denne versjonen håndterer programmet kun reservasjoner for kommende natt. Programmet består av fire klasser du skal skrive (du trenger ikke skrive noe hovedprogram):

Reservasjonssystem. Systemet håndterer ett eller flere hoteller og støtter to typer reservasjoner:

- Reservasjon av et rom med kjent rom-nummer på et navngitt hotell
- Gruppereservasjoner der reservasjonssystemet selv finner og reserverer *passende rom*[1] til så mange som mulig i gruppen, uavhengig av hotell.

Gruppe. En gruppe består av en eller flere personer som ønsker å bestille sammen, og som har felles krav til fasiliteter på rommet/ rommene de ser etter. Personer i en gruppe bryr seg ikke om hvem de blir plassert på rom med så lenge de er i samme gruppe.

Hotell. Et hotell har et navn og en samling gjesterom av klassen **Rom**.

Rom. Hvert rom har et romnummer og et antall senger. Rommet har null eller flere fasiliteter (for eksempel bad eller utsikt). Om rommet er utleid har det en eller flere gjester.

Du velger selv hensiktsmessige instansvariabler utover de som er oppgitt. Instansvariabler skal ikke aksesseres utenfra klassen (de er non-public). Personer representeres kun med navn (en streng som kan inneholde blanke). Fasiliteter og krav til fasiliteter representeres med en tekstlig beskrivelse (en streng uten blanke, for eksempel "utsikt" eller "peis").

Husk å bruke metoder i andre klasser der dette er hensiktsmessig. Du kan anta at returverdier og argumenter til metoder er gyldige, og trenger ikke fange opp eller sjekke for feilsituasjoner.

Oppgave 4a) (3 poeng)

Skriv klassen **Gruppe**. Konstruktøren skal ha én parameter; en liste med krav til fasiliteter i rom. Klassen skal tilby følgende metoder i tillegg til konstruktør:

- **legg_til_personer().** Parameter er en liste med personnavn som legges til gruppen (i tillegg til de som eventuelt var med i gruppen fra før)
- **hent_personer().** Returnerer navnene i gruppen
- **hent_krav()**. Returnerer fasilitetene gruppen krever i sine rom.

Oppgave 4b) (8 poeng)

Skriv klassen **Rom**. Romnummer, antall senger og fasiliteter er parametere til konstruktøren. Klassen skal tilby følgende metoder i tillegg til konstruktør:

- **reserver().** Metoden har en parameter som angir navn på en eller flere gjester som skal bo på rommet. Metoden registrerer at rommet er opptatt og navn på gjesten(e).
- **hent_ant_senger().** Returnerer antall senger i rommet.
- **passer().** Metoden har en parameter for krav til fasiliteter og returnerer True hvis rommet *passer*, ellers False. Metoden tar ikke hensyn til antall senger.
- **__str__().** Returnerer en streng med romnummer og antall senger, samt fasiliteter i rommet.

[1] *passende rom er i denne oppgaven definert som ledige rom som har alle fasiliteter som kreves av en gruppe*

Oppgave 4c) (10 poeng)

Skriv klassen **Hotell**. Hotellnavnet er parameter til konstruktøren, som leser all informasjon om rommene i hotellet fra en fil med navn *hotellnavn.txt* der *hotellnavn* er navnet på hotellet. Hvert rom har en linje i filen. En slik linje inneholder ett heltall for romnummer, ett heltall for antall senger, og null eller flere ord der hvert ord angir en fasilitet. En fil med to rom kan f. eks. se slik ut:

400 4
401 2 bad utsikt peis

Rommene i et hotell lagres i en ordbok der nøklene er romnumre og referanser til objekter av klassen **Rom** er verdier. Klassen skal tilby følgende metoder i tillegg til konstruktør:

- **reserver_rom().** Reserverer et rom for en eller flere gjester. Romnummer og navn på gjesten(e) er parametere.
- **finn_passende_rom()**. Metoden har en parameter som angir krav til fasiliteter, og returnerer en liste med **Rom**-objekter som passer. Metoden tar ikke hensyn til antall senger.

Oppgave 4d) (7 poeng)

I denne og de neste deloppgavene skal du skrive klassen **Reservasjonssystem**. Konstruktøren tar en liste med hotellnavn som parameter, og oppretter en ordbok der hotellnavn er nøkler og referanser til hotell-objekter verdier. Du kan anta at ingen hotell-objekter er opprettet før det opprettes et **Reservasjonssystem**. I denne deloppgaven skal du foruten konstruktøren skrive følgende metode:

- **reserver_rom_i_hotell()**. Metoden reserverer et rom i et hotell. Parametere til metoden er hotellnavn, romnummer og navn på gjesten(e) som skal bo i rommet.

Oppgave 4e) (7 poeng)

I denne deloppgaven skal du skrive en intern (non-public) hjelpemetode i **Reservasjonssystem**:

- **_finn_alle_passende_rom**(). Metoden har en parameter for fasiliteter som kreves og returnerer en ordbok med alle passende rom i alle hoteller. Ordboken har hotellnavn som nøkler og en liste med ledige, passende **Rom**-objekter som verdi for hver nøkkel. Hoteller uten passende rom skal ikke være med i ordboken.

Oppgave 4f) (12 poeng)

I denne deloppgaven skal du skrive metoden i **Reservasjonssystem** som reserverer rom til en gruppe:

- **gruppe_reservasjon().** Metoden har en parameter for gruppen det skal reserveres rom til, og returnerer en bekreftelse på resultatet. Bekreftelsen er en liste med en eller flere strenger.

Dersom ingen rom i noen hoteller passer returneres strengen "Ingen passende rom", og ingen reservasjon utføres. Metoden går ellers gjennom passende rom i alle hoteller og bestiller ett og ett rom til så mange personer fra gruppen som det er senger i rommet. Når alle gjestene har fått rom, eller det ikke er flere passende rom, returnerer metoden en bekreftelse med oversikt over reserverte rom (med hotellnavn og informasjon om rommet). Dersom ikke alle i gruppen har fått rom skal bekreftelsen avsluttes med en advarsel om hvor mange personer som ikke har fått plass.

Oppgave 4g) 3 poeng (teori)

Hva er grensesnittet til klassen **Reservasjonssystem?** Skriv svaret i en kommentar øverst i programmet. Du kan bruke Python-kode, naturlig språk, eller en kombinasjon.