

## Oppgave 7

Herr Glum lager en julekalender til barna sine hvert år. Kalenderen inneholder en gave for hver dag i desember, frem til og med julaften 24.12. Det går på omgang mellom barna hvem som får lov å åpne dagens gave – når alle har åpnet en gave hver, er det førstemann sin tur igjen. Nå ønsker herr Glum seg et program som kan hjelpe ham med holde rede på hvilke barn som får hvilke gaver, og hvor mye gavene hvert barn får, har kostet.

Programmet skal kunne lese inn data om gavene fra en fil som herr Glum oppdaterer etter hvert som han handler inn gaver i tiden før desember. Filen inneholder 24 linjer, en for hver gave: På hver linje står først navnet på gaven (en tekststreng) avsluttet med komma, deretter prisen (et flyttall). Du skal skrive et program i Python som leser inn filen med alle gavene, og deretter hjelper Herr Glum med å holde orden på gavefordelingen.

### 7.1

**Skriv en klasse `Gave`** med to instansvariabler som forteller hva som er i gaven (gavens navn), og hvor mye den har kostet. Klassen skal ha en konstruktør med parametere som angir verdier for instansvariablene. Foruten konstruktøren skal klassens grensesnitt omfatte tre metoder: En som returnerer gavepris; en som returnerer gavenavn; og en metode `__str__` som returnerer gavens navn og verdi som en lesbar **String**.

## 7.2

**Klassen barn** skal ha en datarepresentasjon for barnets navn, alle gavene barnet har åpnet, og totalverdien av gaver barnet har åpnet. Grensesnittet til klassen skal være en konstruktør med barnets navn som parameter, en metode som returnerer barnets navn, en metode som returnerer totalverdien av alle gaver barnet har mottatt, en metode **apneGave** som legger til en ny gave og oppdaterer totalverdien av gaver barnet har fått, og en metode **skrivBarn** som skriver ut på terminal barnets navn, en linje for hver av barnets gaver og til slutt totalverdien av barnets gaver. Skriv klassen **Barn** med alt innhold.

### 7.3

Vi skal utvide programmet med en klasse **Julekalender** med følgende metoder i grensesnittet:

- En konstruktør med parametere for
  - Navn på alle barna i familien
  - Navn på gavefilen der informasjon om gavene ligger lagretKonstruktøren skal opprette objekter for alle barna i familien og opprette selve julekalenderen med gaver.
- En metode **nyDag** som åpner en ny gave og oppdaterer datastrukturen (inkludert hvilken dag og hvilket barn)
- En metode **gaveOversikt** som skriver en oversikt over barna, deres åpnete gaver og totalpris per barn på skjermen

For øvrig har klassen blant annet følgende private innhold:

- En metode **\_lesGavefil** som leser inn gaver med pris fra gavefilen. Filformatet er beskrevet ovenfor.
- En liste **\_kalender** med referanser til en gave for hver dag
- En liste **\_apnere** med referanser til hvert av barna i familien
- Et heltall **\_nesteApner** som holder rede på hvem sin tur det er til å åpne
- Et heltall **\_dag** som holder rede på hvilken dag som skal åpnes neste gang



## 7.4

Vi skal nå utvide klassen Julekalender med historikk fra tidligere år. I første omgang er hensikten å redusere sjansen for at et barn får samme gave flere år på rad. Du kan anta at samme type gave alltid vil ha samme navn, slik at det er lett å sammenligne.

For å representere alle gaver som har vært gitt tidligere, utvider vi klassen Julekalender med en dictionary **\_historikk** der nøkkel er barnets navn, og verdien er en liste med gaver dette barnet har fått. Du kan anta at følgende setninger er lagt til konstruktøren i klassen Julekalender (du trenger ikke endre dette i besvarelsen din):

```
self._historikk = {}
self._lesHistorikk("Historikk.txt")
```

Metoden **\_lesHistorikk** leser data fra en fil der hver linje representerer et barn og har formatet

```
barnenavn gavenavn gavenavn gavenavn gavenavn . . .
```

Du skal nå utvide klassen Julekalender med metoden `_lesHistorikk` (ikke `_skrivHistorikk`):

```
# leser data fra fil til instansvariabelen _historikk
def _lesHistorikk(self, histfilnavn) :

# skriver ut oppdatert historikk (inkludert årets gaver) på fil
def _skrivHistorikk (self, histfilnavn) :
```

## 7.5

Metoden **avvergetLike** i klassen **Julekalender** kalles for å avverge at et barn får en gave det har fått i tidligere år. Den kalles hver dag før metoden `apneGave`. Dersom gaven og barnet som står for tur sammenfaller med noe som er gitt tidligere år, byttes gaven som står for tur med en gave for en senere dag. Hvis metoden ikke klarer å hindre at et barn får en gave det har fått før, skal den returnere **False**. Dersom barnet ikke har fått gaven som står for tur tidligere, eller du klarer å avverge det ved å bytte, skal metoden returnere **True**.

Skriv metoden **avvergetLike** i klassen **Julekalender**

```
# Prøver å avverge at et barn får en gave det har fått tidligere
# Returnerer true om den lykkes, false om den ikke lykkes
```

```
def avvergetLike (self) :
```