Del 4: Julegavefikseren: En kombinert ønskeliste og juleferiekalender

Du kan anta i programmet du skriver at input (argumenter til metodene dine eller input fra fil/bruker) er lovlige verdier av riktig type, du trenger altså ikke sjekke disse om det ikke nevnes spesielt.

Dolly Duck har tenkt ut en løsning på to problemer: Hun er lei av å få julegaver hun ikke ønsker seg og som bare står og støver ned – og hun har ingenting å gjøre i juleferien etter julaften. Løsningen hennes er en app som du skal få utvikle – der hun kan registrere ønsker om opplevelser og aktiviteter i en ønskeliste, og de som vil gi henne gaver kan velge et ønske fra ønskelisten som så legges i en «juleferiekalender».

I juleferiekalenderen kan hun åpne en ny gave hver dag, fra og med 25. desember og så lenge juleferien varer. «Gavene» som ligger i kalenderen er bare en tekst som beskriver opplevelsen eller aktiviteten, sammen med navnet på giveren – men da vet hun jo hvem hun kan kontakte. En vanlig julekalender kan åpnes hver dag fra 1. til 24. desember – men her skal vi altså lage en kalender som brukes *etter* julaften i stedet.

Julegavefikseren er bygget opp av 5 klasser som du skal skrive:

- En klasse Onskeliste som inneholder ønskelisten (en liste med Onske-objekter) og metoder for å legge inn ønsker, velge et ønske å oppfylle, og for å hente ut en liste med ønsker.
- En klasse Onske med metoder for å sjekke om ønsket passer for en giver eller ikke, for å velge et ønske og for returnere ønsket som en string.
- En klasse Juleferiekalender med gaver for hver dag i juleferien, fra og med 25. desember og et valgfritt antall dager fremover (men aldri mer enn 31 dager). Klassen har metoder for å legge til en ny gave, hente gaven for en bestemt dag, og lese av hvor mange dager ferie juleferiekalenderen er laget for.
- En klasse Gave som inneholder beskrivelsen av gaven og hvem den er fra. Denne informasjonen kan hentes ut som en streng.
- Den siste klassen er Julegavefikseren som tar antall dager i juleferien som parameter til konstruktøren. Klassen har metoder for å lese inn Dollys ønsker i ønskelisten fra fil, for å velge et ønske å oppfylle og legge som en gave i juleferiekalenderen (for givere) og for å åpne en ny dag i kalenderen (for Dolly).

Du trenger ikke skrive noe hovedprogram – dette vil lese inn alle ønsker fra fil og ha et brukergrensesnitt der en som vil gi Dolly en gave oppgir navnet sitt og velger et ønske å oppfylle (og hvilken dag det skal gis), og der Dolly senere (når ferien starter) kan «åpne» en ny dag i kalenderen for hver dag i ferien.

Pass på å bruke metoder du allerede har skrevet i andre klasser der det er naturlig. Gave-objekter skal kun refereres til og brukes i Juleferiekalenderklassen, og Onske-objekter skal kun refereres til og brukes i Onskeliste-klassen. Instansvariabler skal aldri aksesseres direkte fra utsiden av klassen de hører til.

Del 4: Julegavefikseren: En kombinert ønskeliste og juleferiekalender

4a) 10 poeng.

Skriv klassen **Onske**. Klassen har instansvariabler beskrivelse, antall (om hun gjerne vil ha ønsket oppfylt flere ganger) og minimumspris (den billigste måten å oppfylle ønsket på). Initialverdiene oppgis i parametere til konstruktøren. Klassen har metodene (som du skal skrive):

- **passer** med parameter som angir maksimumspris. Denne metoden sjekker om et ønske er for dyrt for giveren (minimumsprisen til gaven er høyere enn maksimumskostnaden giveren kan akseptere) eller om ønsket er «brukt opp» (dvs antall = 0)
- **valgt** som kalles når en giver har valgt ønsket. Denne oppdaterer antall og returnerer beskrivelsen
- __**str**__ som returnerer beskrivelse og minimumspris

4b) 10 poeng.

Skriv klassen **Onskeliste** med metoder (som du skal skrive):

- **nytt_onske** med parametere etter behov, oppretter et nytt onske og legger det til onskelisten
- **hent_onsker** som tar maksimumspris (det meste en giver har råd til) som parameter og returnerer en liste med tekststrenger, der hver streng representerer et ønske i ønskelisten. For valgbare ønsker (som kan oppfylles innenfor giverens maksimumspris, og som ikke er «brukt opp») skal strengen være en representasjon av objektet med beskrivelse og pris, mens for ønsker som ikke oppfyller kravene skal det bare stå «Ikke valgbart onske».
- **onske_oppfylles** kalles når en giver har valgt hvilket ønske hen vil oppfylle. Metoden tar som parameter hvilket ønske som er valgt (et heltall), oppdaterer ønsket (at det er «brukt» en gang), og returnerer beskrivelsen av ønsket. Heltallet kan for eksempel være indeks til listen av ønsker.

4c) 5 poeng.

Skriv klassen Gave. Konstruktøren tar parametere med initialverdier til to instansvariabler: beskrivelse og giver (hvem gaven er fra). Klassen har dessuten en metode som returnerer en streng med beskrivelse og giver (som du også skal skrive)

Del 4: Julegavefikseren: En kombinert ønskeliste og juleferiekalender

4d) 10 poeng.

Skriv klassen **Juleferiekalender** med en konstruktør som tar antall dager i kalenderen som parameter og oppretter en ordbok der dagnummer er nøkkel og verdiene foreløpig er None. Dagnummer er et tall som angir datoen i desember eller januar: Fra 25 (siden kalenderen alltid starter 25. desember) til siste dag i ferien (juleferien varer max 31 dager dvs til og med dagnummer 24, men kan også være kortere enn dette). Tallet 28 representerer altså 28.desember, mens tallet 8 representerer 8.januar. Klassen har metodene (som du skal skrive):

- **ny_gave** som oppretter en ny Gave og legger den på en dag i juleferiekalenderen. Parametere er beskrivelse av gaven, giver (hvem gir gaven) og dagen den skal plasseres på (heltallet dagnummer). Du kan anta at den oppgitte dagen er gyldig og trenger ikke sjekke den (selv om dette skulle bety at du overskriver en eksisterende gave).
- **hent_dagens_gave** som returnerer en tekst som skal skrives ut når gaven på en bestemt dag (angitt av parameter dagnummer) åpnes. Teksten skal starte med datoen på formen dagnummer.månedsnavn (for eksempel 25. desember eller 1. januar). Hvis det ikke ligger noen gave i kalenderen for denne dagen skal teksten opplyse om det, ellers skal den inneholde beskrivelsen av gaven og hvem den er fra.
- **hent_ant_dager** returnerer antall dager i denne kalenderen

4e) 7 poeng.

Skriv klassen **Julegavefikser** med en konstruktør som tar antall dager i kalenderen som parameter, og oppretter en Juleferiekalender og en Onskeliste. Konstruktøren skal også initialisere en instansvariabel neste_dag som holder rede på hvilken dag i julegavekalenderen som skal åpnes neste gang. Skriv også metoden:

- **les_onsker_fra_fil.** Metoden tar en parameter filnavn, og leser ett og ett ønske fra filen som har følgende format på hver linje:

  beskrivelse; antall; minimumspris

  For hver linje opprettes et ønske som legges til i ønskelisten.

4f) 8 poeng.

Utvid klassen **Julegavefikser** med metodene (som du skal skrive):

- **velg_gave** leser inn maksimumspris fra en bruker som vil gi en gave, og skriver ut på terminalen en nummerert oversikt over alle ønskene tilpasset denne. Brukeren kan deretter velge et ønske å oppfylle ved å oppgi nummeret, dette registreres i ønskelisten som gitt én gang, og beskrivelsen tas vare på. Til slutt spørres brukeren om hvilken dag i kalenderen gaven skal «legges» i, og en ny gave som oppfyller det valgte ønsket legges der. Du trenger ikke sjekke om det allerede finnes en gave på angitt dag – det er greit at denne i så fall overskrives.
- **ny_dag** kalles når Dolly vil åpne neste dag i kalenderen. Metoden returnerer informasjon om gaven på denne dagen i kalenderen (om det er noen der) og oppdaterer instansvariabelen som holder rede på hvilken dag i kalenderen som skal åpnes neste gang. Husk at etter 31. desember (dagnummer 31) kommer 1. januar (dagnummer 1).