



# Uke 11 - Problemløsning

Gruppe 9 - IN1000  
jonasbny@ifi.uio.no



# Agenda



Problemløsning:

Eksamen h23 oppgave 4

a  
b  
c  
d  
e  
f

# Intro

## Oppgave 4. Garderobe (47 poeng)

I denne oppgaven skal du hjelpe Kim å organisere garderoben sin. Kim har mange klesplagg og vil gjerne få skrevet et program som hjelper til med å sette sammen disse i nye antrekk.

En garderobe er i denne oppgaven samlingen av alle klærne til en person, mens en kategori er en type plagg. Eksempler på kategorier er bukser, skjorter, gensere, og t-skjorter. Alle bukser i garderoben hører for eksempel til i kategorien «bukser». Et plagg hører kun til i én kategori.

Et plagg har minst en farge, og flere plagg kan kombineres til et antrekk. Et antrekk består av plagg som har minst en felles farge, og passer til å brukes sammen i en eller flere anledninger (for eksempel tur eller selskap). Plaggene i et antrekk kommer fra hver sin kategori. For eksempel kan et antrekk bestå av en rød shorts (kategori «bukser») og en hvit t-skjorte (kategori «t-skjorter»). Et annet antrekk kan bestå av en blå langbukse (kategori «bukser»), en hvit skjorte (kategori «skjorter») og en grå og blå dressjakke (kategori «jakker»). Samme plagg kan være med i flere antrekk.

Du skal skrive klassene **Plagg**, **Kategori**, **Garderobe** og **Antrekk** til første versjon av et program til Kim:

- Klassen **Plagg** representerer enkeltplagg i garderoben.
- Klassen **Kategori** representerer en type plagg, og har oversikt over alle plaggene av denne typen i garderoben.
- Klassen **Antrekk** representerer et antrekk: Hvilke plagg som er med i antrekket, og hvilke(n) anledninger antrekket passer til.
- Klassen **Garderobe** organiserer alle plaggene inndelt i kategorier slik at hvert plagg hører til i én (og bare én) kategori. Dessuten har garderoben en liste over antrekk.

Instansvariabler skal ikke aksesseres utenfra klassen (de er non-public). Bruk metoder fra andre deloppgaver der dette er hensiktsmessig (selv om du ikke har løst dem). Du kan anta at returverdier og argumenter til metoder er gyldige, og trenger ikke teste for feilsituasjoner som ikke nevnes.

# Oppgave 4a

## 8 min

4a) 4p Skriv klassen **Plagg** med konstruktør. Konstruktøren tar en parameter farger (liste av string). Klassen skal ha en instansvariabel for en liste av farger og en annen for hvor mange antrekk plagget er med i (int). Skriv også følgende metoder i klassen:

- **har\_farge**. Tar parameteren farge (string). Metoden returnerer True hvis den oppgitte fargen er en av fargene i plagget, ellers False.
- **hent\_ant\_antrekk**. Returnerer antall antrekk plagget er med i.
- **oppdater\_ant\_antrekk**. Oppdaterer hvor mange antrekk plagget er med i. Parameter er endring i antallet (int, kan være negativ).



# Oppgave 4b

18 min

4b) 10p Skriv klassen **Kategori** med konstruktør. Klassen skal ha instansvariabler for kategorinavn (string) og plagg (liste med referanser til plagg-objekter). Kategorinavnet er parameter til konstruktøren. Når kategorien opprettes har den ingen plagg. Videre skal du skrive følgende metoder i klassen:

- **nytt\_plagg.** Parameter skal være farger (liste av string). Metoden oppretter et nytt objekt av klassen Plagg som legges til denne kategorien.
- **finn\_plagg\_med\_farge.** Parameter farge (string). Returnerer liste med 0 eller flere referanser til Plagg-objekter med fargen.
- **trekk\_tilfeldig\_plagg.** Tar en farge som parameter og returnerer en referanse til et tilfeldig Plagg-objekt med riktig farge i kategorien. Om ingen plagg med fargen finnes i kategorien returneres None. Du kan bruke funksjonen **randint(a, b)** som returnerer et tilfeldig tall fra og med a, til og med b, for å trekke tilfeldig (du trenger ikke importere noe i denne besvarelsen).

# Oppgave 4c

15 min

**4c) 7p** Skriv klassen **Antrekk** med konstruktør. Et antrekk skal ha instansvariabler for anledninger (liste av strenger) og plagg (en liste med referanser til Plagg-objekter). Konstruktøren skal ta inn alle plaggene som er med i antrekket (liste av Plagg-referanser) og én anledning (string) som parametere, og oppdatere hvert av plaggene i antrekket for å registrere at de nå er med i et nytt antrekk.

Utvid klassen **Antrekk** med følgende metoder:

- **hent\_plaggene.** Returnerer en liste med referanser til Plagg-objektene i antrekket.
- **legg\_til\_anledning.** Metoden legger til en anledning antrekket passer til. Anledning (string) tas inn som en parameter.
- **passer\_til.** Metoden tar en anledning (string) som parameter og returnerer en boolsk verdi som sier om anledningen er en som dette antrekket passer til.
- **har\_farge.** Metoden returnerer en boolsk verdi som sier om noen av plaggene har en bestemt farge. Fargen (string) er parameter.

# Oppgave 4d

## 15 min

4d) 7p Skriv klassen **Gardero** med konstruktør. Klassen skal ha to instansvariabler:

- en ordbok der kategorinavn (string) er nøkkel og referanse til **Kategori**-objekt er verdi
- en liste med referanser til **Antrekk**-objekter

Ingen av disse initialiseres fra parametere til konstruktøren. Skriv også metoden:

- **nytt\_plagg**. Metoden tar parametere kategorinavn (string) og farger (liste av string) og legger et nytt plagg inn i riktig kategori. Bruk en metode fra en tidligere deloppgave.

# Oppgave 4e

## 17 min

4e) 9p Skriv følgende metoder i klassen **Garderobe** som kan hjelpe Kim å finne passende antrekk:

- **finn\_antrekk\_til\_anledning.** Metoden returnerer en liste med 0 eller flere referanser til alle Antrekk-objekter som passer til en anledning. Anledning (string) er parameter.
- **velg\_første\_antrekk.** Metoden returnerer en referanse til første antrekk den finner som passer til en bestemt anledning og der en bestemt farge finnes på minst et av plaggene. Anledning (string) og farge (string) er parametere til metoden. Finner den ikke et antrekk som passer returneres None.



# Oppgave 4f

## 18 min

4f) 10p Utvid klassen **Garderobe** med følgende metoder som lar Kim sette sammen nye antrekk:

- **finn\_plagg\_til\_antrekk.** Metoden tar en liste med kategorinavn (liste av string) og en farge (string) som parametere. For hver kategori i listen med kategorinavn finner den et tilfeldig plagg med oppgitt farge, og returnerer disse plaggene (liste med referanser til Plagg-objekter). Hvis et av kategorinavnene ikke finnes i garderoben eller kategorien mangler plagg med oppgitt farge, returneres None.
- **lag\_nytt\_antrekk.** Metoden tar en liste med kategorinavn (liste av string), en farge (string) og en anledning (string) som parametere. Dersom den finner plagg med oppgitt farge for hver av kategorinavnene oppretter metoden et nytt antrekk med disse plaggene, legger det til garderoben og returnerer True. Finner den ikke plagg fra hver kategori returnerer metoden False.