

Miniprojekt i Programmering (MIP) for DAT2 og SW2, Forår 2012

Opgaven er delt op i 2 dele. Læs hele opgaven igennem inden I begynder.

1. Struktur

I denne opgave skal der laves et system der håndterer salg af forskellige køretøjer (inspireret af lignende systemer som f.eks. bilbasen.dk).

Systemet skal administrere følgende typer køretøjer: **Personbil**, **lastbil**, **varevogn**, **bus** og **autocamper**. I det følgende beskrives køretøjers fællestræk samt deres forskelle.

Køretøj

Følgende egenskaber er fælles for køretøjer

- **Mærke** (f.eks., "VW" eller "BMW")
- **Model** (f.eks., "Polo" eller "Golf")
 - Såvel mærke som model må ikke være *null*. Forsøg på at tildele en *null* værdi skal udløse en passende exception.
- **Km** (angiver hvor mange kilometer køretøjet har kørt i dets levetid)
 - Km må ikke være et negativt tal. Forsøg på at tildele en negativ værdi skal udløse en passende exception.
- **Årgang** (angiver hvilket år køretøjet er indregistreret)
 - Gyldige årgange inkluderer 1950 – indeværende år.
 - Hvis brugeren forsøger at tildele en værdi udenfor de angivne minimum/maksimum årgange skal systemet automatisk sætte den passende minimum/maksimum værdi.
 - Årgang skal være *read-only*, dvs. efter instantiering må værdien ikke kunne ændres.
- **Pris** (angiver køretøjets salgspris)
- **Max hastighed** (Denne værdi angiver den maskimale tophastighed (i km/t) for en given køretøjstype)
 - Bemærk: Denne værdi angiver ikke hvor hurtigt et konkret køretøj kan køre, men i stedet den maksimale hastighed for alle køretøjer af samme type. F.eks. kan en Fiat Panda køre 167 km/t, mens en Ferrari Enzo kan køre 333 km/t. Begge disse personbiler skal imidlertid have samme max hastighed værdi (407 km/t).
 - For de forskellige køretøjstyper gælder følgende:
 - Personbiler kan max køre 407 km/t
 - Lastbiler kan max køre 95 km/t
 - Varevogne kan max køre 200 km/t

- Busser kan max køre 120 km/t
- Autocampere kan max køre 180 km/t.
- **Motorstørrelse** (som angivet ved slagvolumen i liter)
 - I denne opgave vil forskellige typer køretøjer have forskellige tilladte motorstørrelser:
 - Den tilladte motorstørrelse for personbiler er 1.0 - 4.2 l.
 - Den tilladte motorstørrelse for lastbiler er 5.0 - 15.0 l.
 - Den tilladte motorstørrelse for varevogne er 2.0 - 4.2 l.
 - Den tilladte motorstørrelse for busser er 4.2 - 14.2 l.
 - Den tilladte motorstørrelse for autocampere er 2.4 - 6.2 l.
- **Km/l** (Angiver hvor mange kilometer køretøjet kan køre på en liter brændstof)
- **Brændstof** (Angiver den brændstoftype som køretøjet anvender)
 - Mulige værdier: Diesel, 92 oktan, 95 oktan.
 - Brændstof skal være *read-only* – I skal dog her bruge en anden konstruktion til at angive read-only tilgang end I brugte til Årgang.
- **Energiklasse** (angiver hvor brændstoføkonomisk køretøjet er)
 - Energiklassen er en udregnet egenskab der afhænger af brændstof og km/l egenskaberne.
 - Energiklassen udregnes som udgangspunkt* på følgende måde:
 - Hvis motorens brændstof-type er diesel, er energiklassen:
 - A klasse: Hvis køretøjet kører > 25 km/l
 - B klasse: Hvis køretøjet kører 20-25 km/l
 - C klasse: Hvis køretøjet kører 15-20 km/l
 - D klasse: Hvis køretøjet kører < 15 km/l
 - Hvis brændstof-typen er benzin, er energiklassen:
 - A klasse: > 20 km/l
 - B klasse: 15-20 km/l
 - C klasse: 10-15 km/l
 - D klasse: < 10 km/l
 - * Energiklassen udregnes på en særlig måde for autocampere. Se længere nede.
- **Titel** (Giver en opsummerende beskrivelse af køretøjet)
 - En titel er en udregnet egenskab (dvs., ingen setter) der er sammensat af køretøjets mærke, model og motorstørrelse.
- **Sælger**. Et køretøj er tilknyttet en sælger via en binær association. Et køretøj har således en reference til en sælger, ligesom en sælger har en reference til køretøjet.

Et køretøj skal kunne instantieres på minimum to forskellige måder.

Personbil

- **Veteran** (angiver om personbilen er klassificeret som et veterankøretøj).
 - En personbil klassificeres som et veterankøretøj hvis bilen er mindst 25 år gammel.

- **Titel**
 - Såfremt en personbil er et veteran-køretøj skal det tilføjes til bilens titel.

Lastbil

- **Lasteevne** (Angiver maksimal lasteevne i kg)
- **Soveplads** (Indikerer om lastbilen indeholder en soveplads)
- **Højde** (Angiver køretøjets højde)
- **Vægt** (Angiver køretøjets vægt)

Restriktioner: Den eneste tilladte brændstoftype for lastbiler er diesel. Derudover skal en lastbil indikere at føreren skal have minimum et C-kategori kørekort (stort kørekort). (De samme to restriktioner gælder i øvrigt for busser)

Varevogn

- **Lasteevne** (Angiver maskimal lasteevne i kg)
- **Varerums-dimension** (består af længde, bredde, højde)

Bus

- **Antal siddepladser**
- **Antal sovepladser**
- **Toilet** (Indikerer om bussen indeholder toilet-faciliteter)
- **Højde** (Angiver køretøjets højde)
- **Vægt** (Angiver køretøjets vægt)

Restriktioner: Den eneste tilladte brændstoftype for busser er diesel. Derudover skal en bus indikere at føreren skal have minimum et C-kategori kørekort (stort kørekort). (De samme to restriktioner gælder i øvrigt for lastbiler som beskrevet tidligere)

Autocamper

- **Antal siddepladser**
- **Antal sovepladser**
- **Varmesystem** (Angiver autocamperens primære varmekilde)
 - Mulige værdier er: gas, strøm eller oliefyr.
- **Energiklasse**
 - Autocamperens varmesystem spiller ind på hvilken energiklasse den bliver placeret i. Hvis autocamperen bruger oliefyr, så skal km/l ganges med 0.7 inden energiklassen beregnes.

- Hvis autocamperen bruger strøm, så skal km/l ganges med 0.8 inden energiklassen beregnes.
- Hvis autocamperen bruger gas, så skal km/l ganges med 0.9 inden energiklassen beregnes.
- Eksempel: En benzin-drevet autocamper der kører 21 km/l (klasse A) vil med oliefyr havne i klasse C ($21 * 0.7 = 14.7$), mens en strøm-udgave vil havne i klasse B ($21 * 0.8 = 16.8$)

Sælger

- Navn
- Postnr
- Telefonnummer
 - I skal sørge for at et telefonnummer består af nøjagtigt 8 cifre.
- Hjemmeside
 - Hvis en sælger har angivet en hjemmeside skal i sørge for at den starter med "http://www".
- Køretøjer
 - Det skal være muligt at tilføje og fjerne køretøjer samt gennemløbe en sælgers køretøjer.

En sælger kan være enten en privatperson eller et firma.

Overordnet

Lav et passende klassesdesign der opfylder ovenstående krav, inkl. et passende klassehierarki der reducerer koderedundans. Brug ligeledes passende synlighedsmodifikatorer og datatyper. Navngivning af klasser, egenskaber og metoder står Jer frit for. Ligeledes står det Jer frit for om – og i givet fald hvordan - I vil foretage data validering udover de beskrevne krav.

Alle klasserne i hierarkiet skal override ToString() metoden for at give en fornuftig repræsentation af et givet objekt. Nærmere bestemt, skal et kald til ToString() resultere i at man kan se værdien af alle objektets (offentlige) attributter.

2. Administration af køretøjer

Alle køretøjer skal indkapsles i en passende collection klasse hvorfra køretøjer kan tilføjes, fjernes og gennemløbes. For at undgå duplikker skal I sørge for at et køretøj ikke allerede findes i collection klassen inden det indsættes. Denne undersøgelse skal baseres på køretøjers *værdi-lighed* i stedet for reference-lighed.

Derudover skal klassen implementere følgende søgefunktioner:

- 1) Find alle køretøjer af et bestemt mærke og model.
- 2) Find alle køretøjer der er til salg indenfor et bestemt postnr.
- 3) Find alle køretøjer der har minimum en angivet lasteevne.
- 4) Find alle køretøjer der kræver et stort kørekort, og koster under en bestemt maksimum pris.
- 5) Lav en generel søgemulighed, hvor klienten har mulighed for dynamisk at angive et vilkårligt søgekriterium. Denne søgefunktion vil i sagens natur gøre det muligt for klienter at udtrykke søgefunktionerne 1-4. Forskellen er, at ansvaret for at definere et givet prædikat i dette tilfælde er flyttet over på klienten.

- Køretøjerne der returneres fra søgefunktionerne skal som udgangspunkt returneres i stigende rækkefølge efter pris, dvs. de billigste køretøjer optræder først i resultatet.
- I søgemulighed 3 skal I dog returnere køretøjer i stigende rækkefølge efter deres vægt.

Bemærkninger:

Opgaveløsningen skal kunne demonstreres gennem en klasse der implementerer en simpel tekstuel brugergrænseflade (en konsol-applikation).

Opgaven rummer en del variationsmuligheder mht. opfyldelse af de funktionelle krav, hvorfor der ikke kun findes én rigtig måde at løse den på. Der vil dog være fokus på at I laver et hensigtsmæssigt klassedesign, samt at I sørger for at jeres kode er læsbar.

Ved tvivlsspørgsmål skal I gøre rede for hvilke antagelser der ligger til grund for jeres valgte løsning.

Krav og forventninger til program og dokumentation

- Det udviklede program skal være objekt-orienteret, og det skal skrives i C#.
- Kildeteksten af det udviklede program skal afleveres i to eksemplarer på elektronisk form (CD-ROM, DVD eller USB memory stick). **Blu-Ray medier modtages ikke!**
- Det udviklede program skal være **demonstrerbart**, dvs. det skal kunne afvikles uden bugs.
- Det udviklede program skal være velskrevet og velkommenteret, således at det fremstår forståeligt for læsere af programmet.
- Med det udviklede program skal der følge en kort beskrivelse af programmet (ca. én side). Denne beskrivelse skal omfatte navnene på deltagerne; miniprojekt-navn*; status af programmet; beskrivelse af evt. antagelser, begrænsninger, udvidelser eller variationer i forhold til opgaveformuleringen; underskrift fra alle deltagerne samt eventuelle kilder (dog ikke kursusbogen eller kursuslides). **En underskrift bekræfter deltagelse i programmeringsarbejdet, og at ingen andre end de underskrevne har bidraget til arbejdet.**

- * Miniprojekt-navn: Minigruppen tager navn efter deres oprindelige gruppenavn, f.eks. B126. Første minigruppe har postfiks 'A', mens anden minigruppe har postfiks 'B'. En gruppe B126 vil således blive til to minigrupper, B126A og B126B.
- Evt. uklarheder i opgaveformuleringen forventes afklaret som en del af opgaveløsningen.