

Opgavesæt 2

Når der afleveres løsningsforslag, så pak alle programmets filer, mapper og undermapper – eller projektet som Visual Studio kalder det – ned i én komprimeret fil, altså .suo-filen, .sln-filen plus tilhørende mapper og undermapper..

Man kan hente pakke/komprimeringsprogramet 7-zip gratis her: <http://www.7-zip.org/>, og det kan håndtere de mest brugte komprimerings-formater .zip, .rar og .7z.

Hvis du kan aflevere inden 10. april er det fint

- og du behøver ikke aflevere alle opgaverne. Opgave 2.2 og 2.5 må du gerne springe over, eller gemme til senere ☺

- *ekstraopgaver / udfordringer* forventer jeg **ikke** I afleverer – de er ment som ekstra udfordringer til dem der har kendskab til programmering i forvejen

Opgave 2.1 – Lidt array-gymnastik

- 1) Lav et program der fylder et array af heltal med 50 tilfældige tal mellem 1 og 50, og vis arrayets indhold i et labelfelt.
- 2) Vis arrayets indhold i omvendt rækkefølge i et andet labelfelt.
- 3) Udskriv et givet interval af arrayets elementer i et 3. labelfelt (f.eks. fra element nr. 14 til element nr. 23).
- 4) Udskriv et givet interval af arrayets værdier i et 3. labelfelt (f.eks. alle tal > 10 og < 20).
- 5) Lav en funktion der finder antallet af forekomster af en given værdi i arrayet (f.eks. hvor mange gange værdien 5 forekommer). Funktionen skal som parametre modtage et array og den værdi der skal søges efter, og skal returnere antal forekomster af den søgte værdi. Funktionens signatur skal se sådan ud:

```
public Int32 FindAntal(Int32[] tabel, Int32 tal)
```

- 6) Lav en funktion der finder første forekomst af en given værdi i arrayet (f.eks. hvor værdien 7 første gang findes). Funktionen skal som parametre modtage et array og den værdi der skal søges efter, og skal returnere positionen af den søgte værdi i arrayet. Funktionens signatur skal se sådan ud:

```
public Int32 FindForste(Int32[] tabel, Int32 tal)
```

Angående funktioner, er det en regel, at hvis ikke funktionen finder det søgte returneres værdien -1.

Du kan bruge programeksemplerne ArrayDemo og FunctionDemo som inspiration eller hjælp til at komme i gang med denne opgave.

Opgave 2.2 - Imperial to metric

US Congress gjorde brugen af meter-systemet lovligt i '66 – altså 1866!, men viljen til at "go metric" er en helt anden sag, og diskussionen er nærmest religiøs. England tog det første skridt i starten af 1970'erne da man afskaffede shilling'en, men siden er der ikke sket noget. Der måles stadig i miles, inches, pints, stones osv.

Man kan undre sig over, at de stadig bruger disse enheder, for når vi med det metriske system ubesværet omsætter cm til m til km, er det meget mere besværligt at omsætte inches til yards til miles.

Mange kan måske huske at en inch svarer til 2,54 cm, at en mile er lige godt 1,6 km og at en pint øl er ca. $\frac{1}{2}$ liter. Men hvem ved eller kan huske, hvor mange acres der går på en hektar, eller hvor mange liter der går på en gallon – for slet ikke at nævne, at der er forskel på Imperial- og US-gallons!

Det kunne være praktisk at have et program der konverterer mellem alle disse forskellige målemetoder, så vi f.eks. véd om vi skal kalde Jack Farmer husmand eller storbonde, når vi hører om hans gård på 500 acres.

Længde konverteres således:

- 1 inch (in) = 2,54 cm
- 1 foot (ft) = 12 in
- 1 yard (yd) = 3 ft
- 1 mile = 1760 yd
- 1 nautical mile (sømil) = 2025,4 yd

Areal konverteres således:

- 1 sq inch (in²) = 6,4516 cm²
- 1 sq foot (ft²) = 144 in²
- 1 sq yard (yd²) = 9 ft²
- 1 acre = 4840 yd²
- 1 sq mile (mile²) = 640 acres

Vægt konverteres således:

- 1 ounce (oz) = 28,35 gram
- 1 pound (lb) = 16 oz
- 1 stone = 14 lb
- 1 hundredweight (cwt) = 112 lb
- 1 ton = 20 cwt

Rumfang konverteres således:

- 1 cu inch (in³) = 16,387 cm³
- 1 cu foot (ft³) = 1.728 in³
- 1 fluid ounce (fl oz) = 28,413 ml (Imperial) / 29,574 ml (US)
- 1 pint (pt) = 20 fl oz (Imperial) / 16 fl oz (US)
- 1 gallon (gal) = 8 pt

Inden du kaster dig over opgaven, så stop lige op og gør dig nogle overvejelser om opgaven og del den evt. i grænseflade og funktioner.

Hvordan skal grænsefladen se ud

- ☐ skal der laves 4 skærbilleder/forme, eller en stor GUI evt. opdelt i 4 dele ved brug af tabs, eller måske du har en helt 3. ide?
- ☐ hvilke input skal programmet kunne håndtere. Skal længdemål f.eks. kunne 'inputtes' i mm, cm, m, km, sømil, inch, foot, yard eller mile og programmet så omregne til alle de øvrige?
- prøv at gøre det så fleksibelt som muligt

Hvordan skal funktionaliteten udformes

- ☐ det er vel oplagt at lave et antal funktioner ☺

Jeg glæder mig til at se løsninger, som kan give mig svaret på disse spørgsmål:

- Min bils benzintank rummer 60 liter, gad vide hvor mange gallons det er?
- Min farbror har et mindre landbrug på 25 ha, hvor mange acres svarer det mon til?

Opgave 2.3

Lav et program, der som input modtager et årstal i en tekstboks, og som i et labelfelt viser om årstallet er et skudår eller ej.

Reglen er, at det er skudår hvis 4 går op i årstallet, *undtagen* hvis årstallet er et helt århundrede – *dog* er hvert 400. år skudår. F.eks. er år 1600 skudår, men ikke år 1700, 1800 eller 1900.

Altså en regel med en undtagelse, og desuden en undtagelse til undtagelsen – kalenderen ér bøvlet ☺

Opgave 2.4

Lav et program, der kan beregne billetprisen for at køre med bybusserne.

Grundprisen er 17 kr., og der gives rabat efter følgende regler:

Alder	Rabat
Til og med 4 år	100%
5-15 år	50%
16-59	Ingen rabat
60 år eller ældre	50%

Programmets grænseflade *kunne* bestå af en tekstboks til indtastning af alder, en tryknap til at aktivere beregningen, og et labelfelt til at vise den beregnede billetpris.

Og så en *ekstraopgave / udfordring*:

Når du har fået det til at virke, kan du udvide programmet, så der kan ydes en rabat på 25% til studerende (gælder kun aldersgruppen 16-59). Grænsefladen kunne tilføjes en checkbox hvori kan markeres hvis man er studerende.

Opgave 2.5

Lav et program, der kan benyttes som en simpel valutaregner.

Grænsefladen skal have 3 tekstbokse: én til beløb i danske kroner og en til vekselkursen (prisen i danske kroner for 100 af den udenlandske valuta), desuden en label til beløb i udenlandsk valuta, samt en Beregn-knap og en Slet-knap.

Regneren skal fungere sådan, at når der klikkes på Beregn-knappen, udregnes beløbet i udenlandsk valuta. Når der klikkes på Slet-knappen, skal tekstbokse og label 'tømmes'.

Og så en ekstraopgave / udfordring:

Når beregningen virker kan programmet forbedres lidt, f.eks. ved at

- Lade programmet tjekke om input er korrekt, altså tal (decimaltal) i begge tekstbokse
- Lade Beregn-knappen være 'vissen' indtil der er indtastet i begge tekstbokse
- Lade Slet-knappen være 'vissen' indtil der er indtastet i en af tekstboksene
- ... og måske mere endnu ☺

Opgave 2.6 – Varme drikke

Forestil dig, at du skal programmere en kaffeautomat, der kan levere kaffe, the, cacao og suppe. Hvis der vælges kaffe eller the, skal det være muligt også at få drikken med sukker og/eller fløde. Hvis der vælges cacao, skal det være muligt at få den med flødeskum. Hvis man vælger suppe, skal man ikke kunne få hverken sukker, fløde eller flødeskum.

Der gælder følgende priser:

Tillæg for:

Kaffe:	6,00	sukker:	1,00
The:	5,00	fløde:	1,00
Cacao:	7,00	flødeskum:	2,00
Suppe:	7,00		

Lav et program der kan simulere sådan en kaffeautomat. Grænsefladen kan f.eks. laves med

- Radiobuttons til valg mellem kaffe, the, cacao eller suppe (radiobuttons kan kun vælges én ad gangen)
- Checkbokse til valg af tilbehør - sukker, fløde og flødeskum (checkboxbokse kan vælges til og fra uafhængigt af hinanden)
- En knap til at starte beregningen
- En label til at vise prisen for det valgte