

Opgaver i programmering i C++

1.1

Denne opgave omhandler en klasse til håndtering af EAN (international article number) stregkoder på 13 cifre.

Klassen *EAN* indeholder det 13-cifrede nummer som en attribut med navnet *EANnr* i tekstform (datatype *string*).

Derudover skal klassen som minimum indeholde følgende metoder:

- Default constructor
- Constructor med *string* som eneste parameter til initialisering af *EANnr*.
- Metode til validering af *EANnr* med navnet *isValid*. Der returneres *true*, hvis nummeret er validt, ellers *false*. Valideringen foregår på følgende måde: cifrene kan nummereres fra 0 til og med 12. Cifrene på de ulige positioner (1,3,5,7,9,11) summeres, og summen ganges med tre. Til dette tal adderes summen af cifrene på de lige positioner (0,2,4,6,8,10,12). Hvis det her ved fremkomne tal er deleligt med 10, så er *EANnr* validt.

Nedenstående udskriver 1 (*EANnr* validt).

```
#include "EAN.h";
#include <iostream>
using namespace std;
int main()
{
    EAN e("9780674062313");

    cout << e.isValid();
}
```

1.2

Bøger identificeres ved hjælp af den såkaldte ISBN-kode. Syntaksen for ISBN har ændret sig over tid og varieret fra land til land. Opgaven her går ud på at 'konvertere' et ISBN-nummer med nedenstående format til et validt EAN-nummer:

ISBN 0-13-222220-5

Altså de 4 store bogstaver "ISBN" efterfulgt af én blank, efterfulgt af ét ciffer, efterfulgt af én bindestreg, efterfulgt af to cifre, efterfulgt af én bindestreg, efterfulgt af seks cifre, efterfulgt af én bindestreg og til sidst ét ciffer. Altså en tekststreng, som indeholder i alt 18 karakterer.

Det tilsvarende EAN-nummer er:

9780132222204

Altså en tekststreng, som indeholder 13 numeriske tegn.

Forskellene kan beskrives på følgende måde:

1. EAN indeholder hverken bindestreger eller blank
2. "ISBN" i ISBN-koden er erstattet med "978" i EAN-nummeret
3. Det sidste ciffer i EAN-nummeret er et andet end i ISBN-koden

Det sidste ciffer i EAN-nummeret er et checkciffer, som skal beregnes efter forskrifterne angivet ovenfor.

Du kan vælge at implementere metoden direkte i EAN-klassen med den precondition, at ISBN-nummeret er korrekt. Du kan også vælge at skrive en separat ISBN-klasse med selvstændig validering.

2.

Denne opgave går ud på at programmere en del af domænelaget til en applikation, som skal bruges til afregning af fjernvarmeforbrug.

Først skal der programmeres en klasse ved navn *Forbruger* med (som minimum) følgende indhold:

Attributter:

<i>navn</i>	forbrugerens navn (datatype <i>string</i>)
<i>maalerNr</i>	fjernvarmemålerens nummer. Heltal. Nummeret 'identificerer' objektet.
<i>nyAflaesning</i>	måler værdi (kubikmeter) ved seneste aflæsning. Heltal.
<i>glAflaesning</i>	måler værdi (kubikmeter) ved forrige aflæsning. Heltal.
<i>maalerMax</i>	maksimal mulig værdi på måler. Når måleren passerer denne værdi, starter den forfra ved 0 (overløb). Heltal.

Metoder:

<i>Forbruger</i>	constructor, som kan initialisere attributterne <i>navn</i> , <i>maalerNr</i> og <i>maalerMax</i> .
<i>getMaalerNr</i>	get-metode til <i>maalerNr</i> .
<i>getNavn</i>	get-metode til <i>navn</i> .
<i>setNavn</i>	set-metode til <i>navn</i> (fx til anvendelse ved ny indehaver af lejemål).
<i>aflaesMaaler</i>	som parameter har metoden forbrugerens aflæsning af måler værdien. Metoden skal opdatere <i>nyAflaesning</i> og <i>glAflaesning</i> .
<i>beregnForbrug</i>	returnerer differencen mellem <i>nyAflaesning</i> og <i>glAflaesning</i> , idet der tages højde for eventuelt overløb.

Herefter programmeres klassen *Distrikt* med (som minimum) følgende indhold:

Attributter:

<i>forbrugere</i>	en <i>collection</i> af <i>Forbruger</i> -objekter med plads til præcis 200 forbrugere.
<i>antalForbrugere</i>	antal forbrugere, der er tilknyttet distriktet.
<i>prisM3</i>	pris i kroner pr. kubikmeter fjernvarme. Heltal.

Metoder:

<i>Distrikt</i>	constructor som kan initialisere attributten <i>prisM3</i> .
<i>addForbruger</i>	har <i>Forbruger</i> -objekt som parameter, og dette tilføjes collection.
<i>afregnForbruger</i>	har <i>maalerNr</i> som parameter og returnerer prisen på fjernvarmeforbruget. Hvis måleren ikke findes, returneres -1.

Nedenstående test-driver kan anvendes til at teste din applikation.

```
int main()
{
    Distrikt svindinge(78); //Opretter distrikt med kubikmeterpris 78 kr.

    Forbruger poul("Poul Thorsen", 1206, 999);
    Forbruger niels("Niels Kragh Hansen", 1227, 999);

    svindinge.addForbruger(poul);
    svindinge.addForbruger(niels);

    poul.aflaesMaaler(924);
    poul.aflaesMaaler(18);

    niels.aflaesMaaler(608);
    niels.aflaesMaaler(724);

    cout << svindinge.afregnForbruger(1206) << endl; //Udskriver 7332
    cout << svindinge.afregnForbruger(1227) << endl; //Udskriver 9048
    cout << svindinge.afregnForbruger(1256) << endl; //Udskriver -1
}
```