

Prøve eksamen 20. april 2018

Indledning

Dette er et skriftligt eksamenssæt, som i struktur og indhold *minder om* den rigtige eksamen på datamatiker 1. semester¹. Der tages højde for fejl og mangler i opgavesættet, da det ikke er kvalitetssikret i samme omfang som et rigtigt eksamenssæt. ☺

I lighed med den ordinære eksamen har alle opgaver tilknyttede JUnit tests, som skal afvikles succesfuldt, for at en opgave er løst tilfredsstillende. I skal altså ikke selv skrive JUnit tests, men teste jeres løsninger op imod de allerede producerede unit tests. Der må IKKE foretages ændringer i de udleverede unit test².

I får udleveret et NetBeans projekt, som indeholder kode (klasser og interfaces), der *skal* bruges i til løsning af opgaverne. Koden og de tilknyttede unit tests er placeret i packages med samme navn som den tilhørende opgave. Eksempelvis ligger kode og unit tests til "Opgave 1" i package med navnet: opgave1.

Materiale

Netbeans projektet hedder `TrialExam` og skal downloades som en zip fil via Sharepoint mappen "Eksamen". Dernæst skal I importere zip fil til jeres egen Netbeans.

Det er ikke en study point opgave, men I bedes alligevel afleverer jeres besvarelse på Moodle.

Opgaverne

Opgave 1 (15%)

a. (5 %)

Implementér metoden `isLeapYear()` i klassen `LeapYear`, der tester for skudår. Metoden modtager et årstal som en `int` og returnerer `true`, hvis året er skudår, og ellers `false`.

Regler for beregning af skudår kan findes her: <https://da.wikipedia.org/wiki/Skud%C3%A5r>

b. (10 %)

Skriv en ny metode i klassen `LeapYear`, der kan returnere antal skudår inden for en periode. Metodens signatur skal se således ud:

```
public int numberOfLeapYears(int start, int end)
```

Startår må ikke være større end slutår. I denne situation returneres -1 som fejlkode.

¹ Da alle eksamensemner ikke har været gennemgået endnu, indeholder opgaverne fx ikke `Exceptions`, som ellers ville være en naturlig del af løsningen på nogen af opgaverne (fejlsценарierne).

² . I forbindelse med vurdering af jeres besvarelser til den ordinære eksamen, skal I forvente, at jeres løsninger vil blive afprøvet med et yderligere antal test cases.

Opgave 2 (10 %)

Skriv en metode i klassen `CalendarCheck`, som implementerer `CalendarCheckInterface`.

Metoden skal undersøge om en dato er en fredag. Metoden modtager et `LocalDate` objekt³, som indeholder en dato, og returnere `true`, hvis ugedagen er en fredag, og ellers `false`.

Hint:

Klassen `LocalDate` har en metode `getDayOfWeek()` som returnerer en ugedag fra en enum kaldet [`DayOfWeek`](#).

`DayOfWeek` repræsenterer de 7 ugedage `MONDAY` til `SUNDAY`.

Man kan fx foretage flg. operation (kun eksempel og ikke direkte overførbart til jeres løsning), som vil udskrive "THURSDAY" på skærmen:

```
System.out.printf(DayOfWeek.MONDAY.plus(3));
```

Opgave 3 (25 %)

Skriv i hver delopgave en metode i klassen `Shop`, som implementerer `ShopInterface`.

a. (5 %)

Implementér flg. metode, som opretter og returnerer et `Customer` objekt:

```
public Customer makeCustomer(String name, String email)
```

Du bør i metoden sikre, at inputparametre indeholder valide værdier (se nærmere i unit tests).

b. (5 %)

Implementér flg. metode, som opretter og returnerer et `Sale` objekt:

```
public Sale makeSale (String description, double price, int qty)
```

Du bør i metoden sikre, at inputparametre indeholder valide værdier (se nærmere i unit tests).

³ `LocalDate` formatet er year-month-day, fx "2018-04-20".

1. Semester Datamatiker

c. (5 %)

Implementér flg. metode, som knytter et Customer objekt til et Sale Objekt (Hvis Sale objektet allerede har et kundeobjekt tilknyttet, knyttes det ny Customer objekt ikke til Sale objektet og der returneres false, ellers true):

```
public boolean addCustomerToSale(Sale sale, Customer customer);
```

d. (5 %)

Implementér flg. metode, som returner prisen på et salg (pris * quantity):

```
public double getSaleTotal(Sale sale);
```

e. (5 %)

Implementér flg. metode, som returner den fulde information om et salg, inkl. kundeoplysninger:

```
public String getFullSale(Sale sale);
```