

Westerdals Oslo ACT

Skriftlig prøve

PG2100 – Programmering 2

Tillatte hjelpeemidler: ingen

Dato: 3.6.15

Vedlegg som kan være aktuelle: 12 (side 5 – 18)

Tid: 180 minutter

I alle oppgavene teller hvert delspørsmål likt dersom ikke annet er oppgitt.

NB! Hvis du synes noe er uklart eller at opplysninger mangler, må du gjøre egne begrunnde antagelser/forutsetninger, og løse oppgaven ut fra disse.

Oppgave 1 (35 %)

Anta at følgende klasser er deklarert (i samme package):

```
public class Sawyer extends Jack {  
    public void follow() {  
        System.out.print("sawyer-F ");  
        super.follow();  
    }  
  
    public class Kate {  
        public void lead() {  
            System.out.print("kate-L ");  
        }  
  
        public void follow() {  
            System.out.print("kate-F ");  
        }  
  
        public String toString() {  
            return "kate";  
        }  
    }  
  
    public class Jack extends Locke {  
        public void lead() {  
            super.lead();  
            System.out.print("jack-L ");  
        }  
  
        public String toString() {  
            return "jack";  
        }  
    }  
  
    public class Locke extends Kate {  
        public void follow() {  
            lead();  
            System.out.print("locke-F ");  
        }  
    }  
}
```

Følgende program er også gitt (ligger i samme package som den nevnt over):

```
public class Oppgave1a {  
    public static void main(String[] args) {  
        Kate[] characters = { new Locke(), new Jack(), new Sawyer(), new Kate() };  
        for (int i = 0; i < characters.length; i++) {  
            System.out.println(characters[i]);  
            characters[i].lead();  
            System.out.println();  
            characters[i].follow();  
            System.out.println();  
            System.out.println();  
        }  
    }  
}
```

a) (10 %) Vis hvordan utskriften blir når programmet over blir utført.

(Oppgaven fortsetter på neste side.)

b) (10 %) Skriv en static metode count som har to parametere:

en ArrayList med Strings (strenger)
et heltall

Metoden skal telle opp hvor mange av strengene i listen som inneholder like mange tegn som verdien av heltallsparameteren. Dette antallet skal returneres.

Hvis heltallsparameteren har verdien 4, og listen inneholder strengene

"this" "is" "lots" "of" "fun" "for" "every" "Java" "programmer"

skal metoden returnere 3. Det er 3 strenger ("this" "lots" "Java") som inneholder 4 tegn (tegnet " er ikke en del av strengene).

c) (15 %) Skriv en static metode `replace` som har en `ArrayList` med strenger som parameter. Metoden erstatter hver streng i listen med en streng av "stjerner" ("*****"). Det skal være like mange stjerner som det er tegn i strengen som skal erstattes. Metoden skal ikke returnere noe.

Hvis metoden kalles med en arraylist som inneholder strengene

"this" "is" "lots" "of" "fun" "for" "every" "Java" "programmer"

skal strengene i listen etterpå være

" * * * * " " * * " " * * * * " " * * " " * * * " " * * * " " * * * * " " * * * * " " * * * * * * * * * "

Oppgave 2 (35 %)

a) (25 %) Skriv en klasse PhoneCard som representerer et forhåndsbetalt kontantkort som holder rede på beløpet som bruker har å ringe for.

(5 %) Klassen skal ha attributter for

beløpet det kan ringes for (credit – et desimaltall)
prisen per ringeminutt (charge – et desimaltall)
en unik identifikasjon (id – en String)
antall minutter det er ringt for totalt (minutes – et desimaltall)

(10 %) Klassen skal bl.a. ha

to konstruktører
standard tilgangsmetoder
en `toString`-metode
en `equals`-metode

Den ene konstruktøren skal ikke ha parametere. Den andre konstruktøren skal ha parametere for de tre første attributtene – antall minutter det er ringt for totalt skal settes til 0.

(10 %) Klassen skal også ha metoden

`call` med en parameter for et antall ringeminutter.

Dette antallet skal metoden gange med prisen per ringeminutt (`charge`), og det resulterende beløpet skal så trekkes fra `credit`. Hvis `credit` ikke er stor nok til å dekke det ønskede antallet ringeminutter, skal metoden beregne hvor mange minutter `credit` dekker, og endre berørte attributter ut fra dette. Metoden skal returnere forbruket (i kroner) for denne samtalen.

b) (10 %) Skriv en klasse PhoneCardProgram med en `main`-metode som oppetter et PhoneCard–objekt ved bruk av konstruktøren med parametere. Programmet skal bruke klassens metoder på passende måte, slik at det kan lage output til konsollvinduet som viser hva som skjer – som vist under.

```
Constructs a phone card (credit = 200 kr, minute charge = 0.5 kr, id = EasyPhoning 1234)
Makes a 10 minutes call
Used kr. 5.0
Credit left: 195.0
Minutes used (total): 10.0

Makes a 5 minutes call
Used kr. 2.5
Credit left: 192.5
Minutes used (total): 15.0

Makes a 1000 minutes call
Used kr. 192.5
Credit left: 0.0
Minutes used (total): 400.0

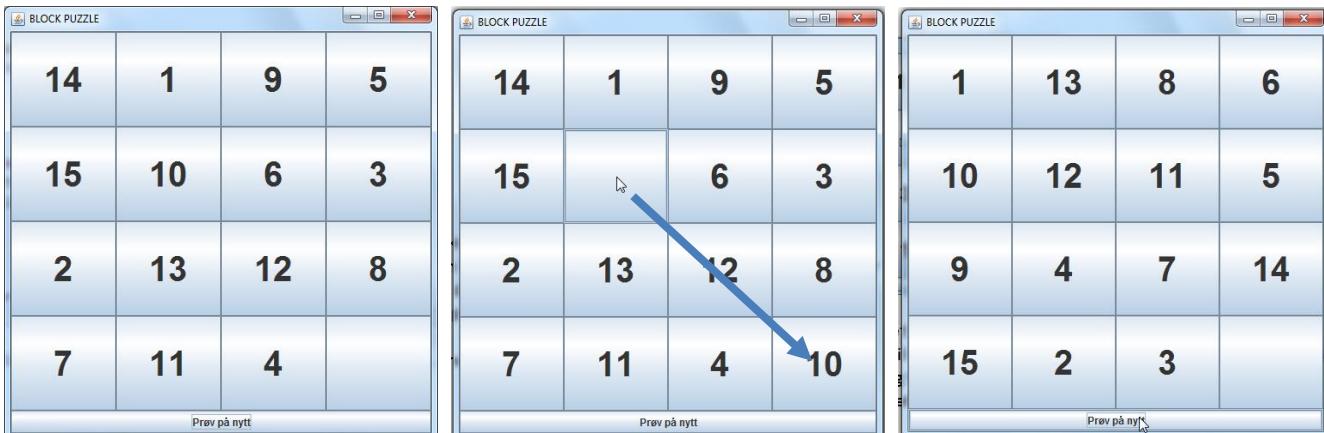
Card info:
Credit left: 0.0
Minute charge: 0.5
Minutes used: 400.0
Id: EasyPhoning 1234
```

Legg merke til at Credit left ble 0 da det ble (forsøkt) gjort en samtale på 1000 minutter. Antall minutter ble beregnet ut fra hva som var igjen av credit på dette tidspunktet.

Oppgave 3 (30 %)

Vedlegg 12 viser programmet BlockPuzzle. Dette setter opp et grafisk grensesnitt som vist i figuren til venstre under.

Du skal skrive noe av koden for dette programmet.



Vinduet inneholder 4 x 4 knapper organisert i et panel, samt en enkelt knapp nederst.

Ved klikk på en av de 16 knappene, flytter tallet på denne knappen seg til knappen uten tall – se figuren i midten over. Der har bruker klikket knappen med tallet 10, og tallet flyttet seg til knappen uten tall.

Ved klikk på knappen "Prøv igjen", settes brettet opp på nytt – som i figuren til høyre over.

Vinduet har størrelsen 500 x 500 pixler.

I koden i vedlegg 12 mangler en del kode, blant annet:

- array's er ikke opprettet
- panelet er ikke opprettet
- layout er ikke satt
- knapper er ikke plassert i panelet
- lytting er ikke implementert for knappene
- oppsett av vindu mangler

Det er markert fire steder i vedlegg 12 med en kommentar (//1, //2 osv.) der det kan mangle kode.

Din oppgave er å skrive koden som du mener mangler på disse stedene, slik at programmet virker som tiltenkt. I din besvarelse skal du bare skrive det du mener mangler, og du skal markere det med de tilsvarende kommentarene som er i vedlegget.

Eksempel:

//1 Her skalstå ...

//2 Her skalstå ...

//3 Her skalstå ...

//4 Her skalstå ...

--- Slutt på oppgavesettet ---