

i Informasjon om eksamen



EKSAMEN

Emnekode: ITF10611

Emnenavn: Objektorientert programmering

Dato: 6. mai 2019

Eksamenstid: 4 timer

Hjelpemidler: To A4-ark (fire sider) med egne notater

Faglærer: Lars Emil Knudsen

Om eksamensoppgaven og poengberegning:

På hver hovedoppgave er det angitt hvor mye disse teller av totalen. Karakter fastsettes på grunnlag av en helhetsvurdering av besvarelsen.

I oppgavene hvor du blir bedt om å skrive kode anbefales det at du skriver løsningen med java- syntaks. Er du derimot usikker på hvordan du skal besvare en oppgave med kode kan du skrive svaret med egne ord, det er da viktig at du beskriver logikken på en omfattende og detaljert måte.

Om en oppgave virker å forutsette at du har løst en tidligere oppgave så kan du forutsette at denne er løst uavhengig om du faktisk har gjort det eller ei. F.eks: om du i en oppgave må benytte metoden: metode() som du skulle ha implementert i en tidligere oppgave, så kan du bare anta at dette er gjort og den fungerer slik det er tenkt.

Pass også på å svare på alle oppgaver. Det er bedre å skrive litt i grove trekk hvordan du ser for deg at oppgaven kan løses, enn å ikke skrive noe i det hele tatt, dersom du står fast.

Takk for et veldig hyggelig semester. Lykke til med videre eksamener og studieløp. :)

Lykke til!

Sensurfrist: 27. mai 2019

Karakterene er tilgjengelige for studenter i Studentweb.

i **Del 1 - Teori - 20%**

Del 1 består av 20 spørsmål. Oppgavene består av teorioppgaver i flersvarsform med fire alternativer. Det er ett alternativ som er korrekt for hver oppgave. Riktig svar gir 3 poeng, feil svar gir -1 poeng og hvis du ikke svarer får du 0 poeng.

1 **Oppgave 1.1**

Konstruktør brukes til å...

Velg ett alternativ

- Lage subklasser
- Initialisere nye objekter
- Frigi RAM
- Bygge grafisk brukergrensesnitt

Maks poeng: 3

2 **Oppgave 1.2**

En abstrakt metode er...

Velg ett alternativ

- En metode som er arvet fra superklassen
- En metode som kan kalles, men ikke gjør noe - den bare returnerer
- En metode som kaster unntak dersom den kalles
- En metode uten "kropp", den har bare "hode"

Maks poeng: 3

3 Oppgave 1.3

Hvilke av disse er korrekt?

Velg ett alternativ

- Statiske variabler kan ikke aksesseres fra en ikke-static kontekst
- Instansvariabel kan ikke aksesseres fra static kontekst
- Instansvariabel kan ikke aksesseres fra ikke-static kontekst
- Statiske variabler kan ikke aksesseres fra en static kontekst

Maks poeng: 3

4 Oppgave 1.4

Hvilket nøkkelord benyttes for å innkapsle instansvariabler og metoder...

Velg ett alternativ

- ingen av de andre
- public
- package
- private

Maks poeng: 3

5 Oppgave 1.5

Hvilke klasstyper kan ikke i noen tilfelle instansieres?

Velg ett alternativ

- Indre klasse
- Nestet klasse
- Abstrakt klasse
- Anonym-indre klasse

Maks poeng: 3

6 Oppgave 1.6

Hvis du i en konstruktør, ønsker å kalle en annen konstruktørm metode i samme klasse benytter du...

Velg ett alternativ

- this
- super
- klassenavnet
- static

Maks poeng: 3

7 Oppgave 1.7

En *indre klasse* er...

Velg ett alternativ

- En klasse som både har superklasse og subklasser
- En klasse definert i en annen klasse
- En klasse som har nøkkelordet "inner"
- En klasse definert i en metode

Maks poeng: 3

8 Oppgave 1.8

Dersom en metode reimplementeres i en subklasse bør metoden merkes med...

Velg ett alternativ

- @Override
- @Overload
- @Reimplement
- @Nullable

Maks poeng: 3

9 Oppgave 1.9

En klasse som *ikke* er abstrakt kan *ikke* inneholde...

Velg ett alternativ

- Abstrakte metoder
- Ikke abstrakte metoder
- Statiske variabler
- Instansvariabler

Maks poeng: 3

10 Oppgave 1.10

Et interface kan inneholde...

Velg ett alternativ

- Konstruktør
- Instansmetoder
- Statiske metoder
- Instansvariabler

Maks poeng: 3

11 Oppgave 1.11

En klasse kan i Java...

Velg ett alternativ

- Arve fra flere klasser, men bare implementere ett interface
- Arve fra én klasse og implementere flere interface
- Arve fra flere klasser og implementere flere interface
- Arve fra én klasse og implementere ett interface

Maks poeng: 3

12 Oppgave 1.12

Hva beskriver best et objekt?

Velg ett alternativ

- En primitiv variabel
- En instans av en metode
- En instans av en klasse
- Barn av en klasse

Maks poeng: 3

13 Oppgave 1.13

De tre hoveddelene av et UML-klassediagram inneholder...

Velg ett alternativ

- Klassenavn, data og metoder
- Klassenavn, static elementer og metoder
- Klassenavn, static-elementer og instanse-elementer
- Klassenavn, private elementer og public elementer

Maks poeng: 3

14 Oppgave 1.14

Dersom en klasse bare har private data og ingen set-metoder, er den da immutable?

Velg ett alternativ

- Ja
- Nei
- Det kommer an på - get-metoder kan avgjøre
- Det kommer an på - static-metoder kan avgjøre

Maks poeng: 3

15 Oppgave 1.15

Hvis en *metode* i en klasse har samme navn, men forskjellige parametere, kalles det...

Velg ett alternativ

- Overloading
- Innkapsling
- Arv
- Overriding

Maks poeng: 3

16 Oppgave 1.16

Forholdet mellom en subklasse (K) og en superklasse (L), kan best beskrives som at:

Velg ett alternativ

- K "er en" L
- L "har en" K
- K "har en" L
- L "er en" K

Maks poeng: 3

17 Oppgave 1.17

En Controller i JavaFX er som regel knyttet til logikken i en...

Velg ett alternativ

- Stage
- Scene
- Scene Graph
- Node

Maks poeng: 3

18 Oppgave 1.18

Hvilken egenskap i OOP illustrerer best gjenbruk av kode?

Velg ett alternativ

- Innkapsling
- Arv
- Polymorfisme
- Primitivitet

Maks poeng: 3

19 Oppgave 1.19

Hvilken av de følgende datatypene kan holde på det største flyttallet?

Velg ett alternativ

- long
- double
- byte
- float

Maks poeng: 3

20 Oppgave 1.20

For å gjøre en naturlig sortering av eliste med objekter med Collections.sort() må en klasse implementere interfacet...

Velg ett alternativ

- Collections
- Comparator
- Comparable
- Equals

Maks poeng: 3

21 Del 1 - Kommentar

Her kan du i **spesielle** tilfeller skrive en kommentar i forhold til flersvarsoppgavene i Del 1 hvis det er oppgave(r) du mener er feil eller trenger en utdypning. Denne gir ingen poeng.

Hvis du ikke har noe, kommenter "-" for å slippe en advarsel fra Inspira om at du har ubesvarte oppgaver.

Skriv ditt svar her...

Maks poeng: 0

i Del 2 - Kodeforståelse - 20%

Oppgavene i del 2 tar for seg analyse og forståelse av Javakode og dens oppbygning, oppgavene er i flersvarsform. Det er ett alternativ som er korrekt for hver oppgave. Riktig svar gir 3 poeng, feil svar gir -1 poeng og hvis du ikke svarer får du 0 poeng.

22 Oppgave 2.1

Hvilke OOP prinsipper vises i følgende kode.

Velg ett alternativ

- Arv og polymorfi
- Polymorfi
- Innkapsling og arv
- Arv

Maks poeng: 3

23 Oppgave 2.2

Hva blir utskriften fra dette programmet?

Velg ett alternativ

- 200
- 20
- 100
- 1000

Maks poeng: 3

24 Oppgave 2.3

Hva blir utskriften fra dette programmet?

Velg ett alternativ

- Vanlig dyr, Vanlig dyr, Vanlig dyr,
- Vanlig dyr, Honninggrevling, Honninggrevling,
- Honninggrevling, Honninggrevling, Honninggrevling,
- Vanlig dyr, Vanlig dyr, Honninggrevling,

Maks poeng: 3

25 Oppgave 2.4

Hva blir utskriften fra dette programmet?

Velg ett alternativ

- [7, 5, 3, 2]
- [2, 3, 5, 7]
- [2, 5, 7, 3]
- Kompileringsfeil

Maks poeng: 3

26 Del 2 - Kommentar

Her kan du i **spesielle** tilfeller skrive en kommentar i forhold til flersvarsoppgavene i Del 2 hvis det er oppgave(r) du mener er feil eller trenger en utdypning. Denne gir ingen poeng.

Hvis du ikke har noe, kommenter "-" for å slippe en advarsel fra Inspira om at du har ubesvarte oppgaver.

Skriv ditt svar her...

Maks poeng: 0

i **Del 3 - Programmering - 60%**

Du er med på et prosjekt som skal lage et avspillingsprogram. Først skal det utvikles en klasse for å representere lyd. Klassen skal ha instansvariabler for:

- Navnet på et lydklipp
- Opptakets varighet i sekunder
- Antall kanaler – 1 for mono, 2 for stereo osv. opp til 24
- Samplingsrate – antall «målinger» av lydbilde per sekund (ofte 44100)
- Antall bit som lagres for hver måling (hvert sample), typisk 8 – 24
- Selve samplene – alle målingene som er gjort av lydbildet. Her kan skallet av klassen `LydSamples` som ligger ved som vedlegg benyttes (noen andre skal gjøre ferdig implementasjonen av denne klassen).

Klassen skal ha en konstruktørmetode som tar imot alle seks verdiene. Test om antall kanaler er rimelig. De andre parameterne behøver ikke testes i denne omgang. Ved feil skal det kastes en `IllegalArgumentException`. Videre skal klassen ha get-metoder for alle instansvariablene, men ingen set-metoder.

27 Oppgave 3.1

Lag klassen «Lyd» basert på de opplysningene du har fått.

"Klassen skal ha instansvariabler for:

- Navnet på et lydklipp
- Opptakets varighet i sekunder
- Antall kanaler – 1 for mono, 2 for stereo osv. opp til 24
- Samplingsrate – antall «målinger» av lydbilde per sekund (ofte 44100)
- Antall bit som lagres for hver måling (hvert sample), typisk 8 – 24
- Selve samplene – alle målingene som er gjort av lydbildet. Her kan skallet av klassen LydSamples som ligger ved som vedlegg benyttes (noen andre skal gjøre ferdig implementasjonen av denne klassen).

Klassen skal ha en konstruktørmetode som tar imot alle seks verdiene. Test om antall kanaler er rimelig. De andre parameterne behøver ikke testes i denne omgang. Ved feil skal det kastes en `IllegalArgumentException`. Videre skal klassen ha get-metoder for alle instansvariablene, men ingen set-metoder."

Skriv ditt svar her...

1	
---	--

Maks poeng: 10

28 Oppgave 3.2

Følgende metoder skal også være en del av Lyd-klassen:

```
public void spill()  
public long getSize()  
public boolean cdKvalitet()
```

Lag implementasjonene for getSize() og cdKvalitet(). Metoden spill() skal noen andre på prosjektet lage, den trenger du ikke implementere.

```
public void spill()  
Denne starter avspilling av lyd (selve implementasjonen lages av noen andre)
```

```
public long getSize()  
Denne returnerer objektets størrelse målt i byte. Det finner man ved å multiplisere antall kanaler, samplingsrate, bytes per sample, og varighet. En sample fyller et helt antall byte, så dersom en sample bruker 20 bits så trengs det 3 bytes.
```

```
public boolean cdKvalitet()  
Denne returnerer true bare dersom lyden har minst to kanaler, minst 44100 samples per sekund (også kalt hertz – Hz), og minst 16 bits sample.
```

Skriv ditt svar her...

1	
---	--

Maks poeng: 10

29 Oppgave 3.3

Etter at lydklassen er på plass kommer det inn et forslag om at systemet også skal støtte video, det trengs derfor en «Video»-klasse. Etter en diskusjon kommer dere til at lydklassen og videoklassen har noen felles egenskaper som skal beskrives i en egen superklasse «Spillbar». De felles egenskapene er at både lyd og video har et navn, begge kan spilles, begge har en viss varighet.

Lag superklassen «Spillbar» i samsvar med beskrivelsen, med de instansvariabler og metoder den bør ha. Vurder om noe bør være abstrakt. Vis/forklar hvordan klassen «Lyd» må endres for å benytte denne superklassen.

Skriv ditt svar her...

1	
---	--

Maks poeng: 10

30 Oppgave 3.4

«Video»-klassen skal nå lages. I tillegg til de tidligere nevnte egenskapene, skal den også inneholde to instansvariabler til lyd og bildesekvens. For lyd bruker du klassen «Lyd» som er laget. For bildesekvensen bruker du en ferdig klasse «BildeSekvens» noen andre på prosjektet skal lage.

Klassen «Video» skal ha en konstruktørmetode som tar tre parametere, navn, lyd og bildesekvens. Videoen sin varighet kan settes lik den største av lyd og bildesekvens sine varigheter.

Spill-metoden trenger ikke implementeres.

Skriv ditt svar her...

1	
---	--

Maks poeng: 10

31 Oppgave 3.5

Det vil være aktuelt å kunne sortere både lydklipp og videoer. Vis nødvendige endringer i en eller flere klasser som gjør en naturlig sortering på navn mulig.

Skriv ditt svar her...

1	
---	--

Maks poeng: 10

32 Oppgave 3.6

For å presentere lydklipp og videoer i applikasjonen ønsker vi i første omgang å bruke toString mens vi utvikler. Den skal returnere en String som starter med «Lyd - » for lyd eller «Video - » for video, etterfulgt av navn, og varighet i sekunder. Vis hvilke utvidelser du vil gjøre i klassene.

Skriv ditt svar her...

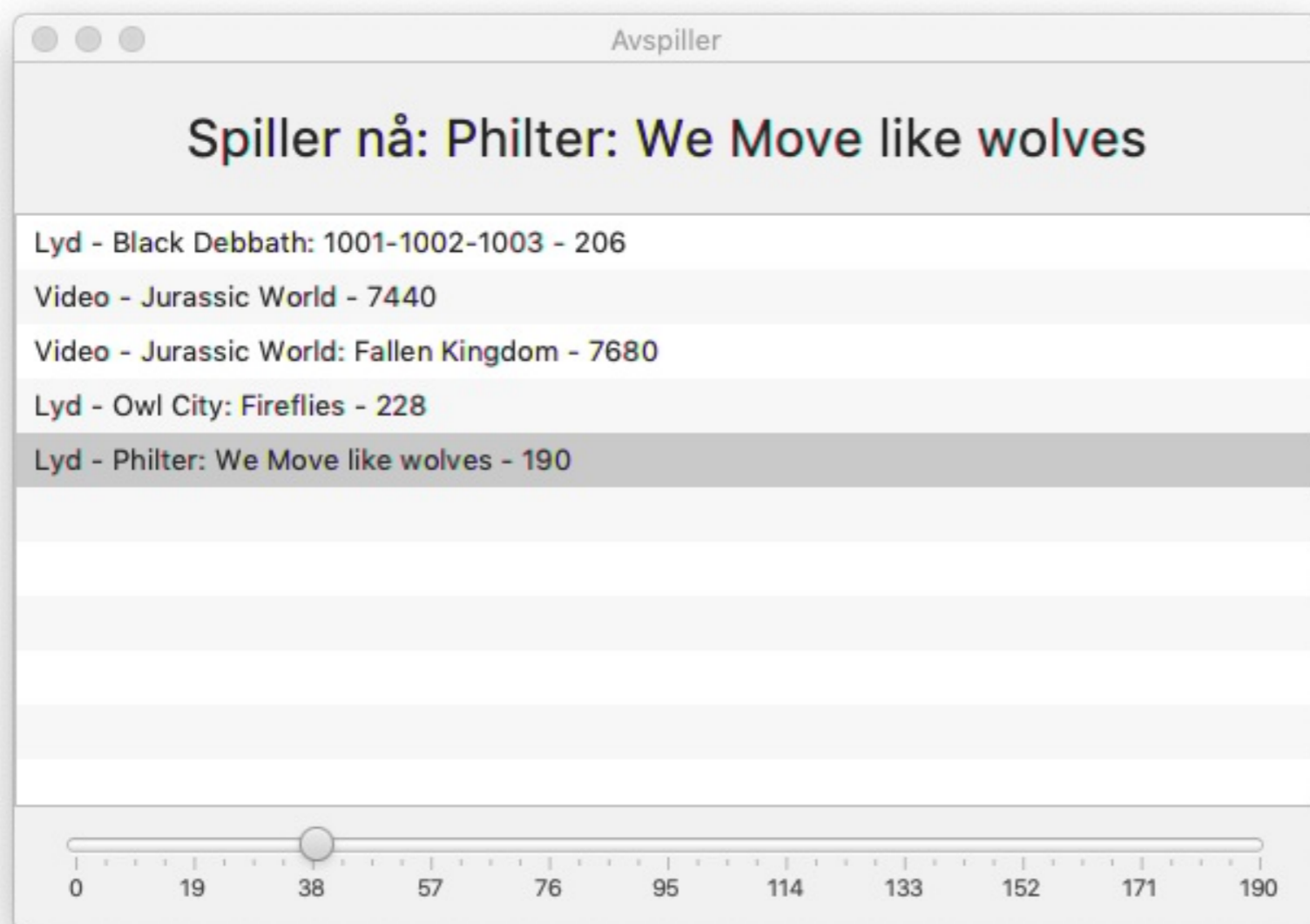
1	
---	--

Maks poeng: 10

33 Oppgave 3.7

Under utviklingen lages et enkelt grensesnitt for å teste avspillingen. Selve grensesnittet i form av en «Avspiller.fxml» er laget, men du skal fylle ut noe av funksjonaliteten som ligger i AvspillingsController-en som ligger ved som vedlegg.

Du skal legge inn to objekter til i ListView'et, ett Lyd-objekt og ett Video-objekt. Gå ut ifra at du kan benytte en tom-konstruktør for Lyd og BildeSekvens når du lager Video-objektet. Sorter også listen på naturlig orden.



Skriv ditt svar her...

1	
---	--

Maks poeng: 10

34 Oppgave 3.8

Det er lagt til en ChangeListener til ListView'et. Sørg for at status-teksten og slideren blir oppdatert ved et valg i listen.

Status-teksten skal inneholde «Spiller nå: », etterfulgt av navn.

Slideren skal gis en maksgrense lik varigheten til det valgte elementet og det skal vises 10 «tick»-markere. I tillegg skal selve Slideren sin verdi være på 30% av maksgrensen.

Et utdrag av JavaDoc til Slider-klassen ligger vedlagt.

Skriv ditt svar her...

1	
---	--

Maks poeng: 10

Question 22
Attached



OPPGAVE 2.1

```
public class Sykkel {
    private String type;
    private int gir;

    public Sykkel(String type, int gir) {
        this.type = type;
        this.gir = gir;
    }
}

public class TerrengSykkel extends Sykkel {
    private int seteHoyde;

    public TerrengSykkel(String type, int gir,
                          int seteHoyde) {
        super(type, gir);
        this.seteHoyde = seteHoyde;
    }

    public int getSeteHoyde() {
        return seteHoyde;
    }
}
```

Question 23
Attached



OPPGAVE 2.2

```
public class Kalkulator {
    private int tall = 100;

    public void kalkuler(int tall) {
        this.tall = tall * 10;
    }

    private void printTall() {
        System.out.println(tall);
    }
}

public static void main(String[] args) {
    Kalkulator kalkulator = new Kalkulator();
    kalkulator.kalkuler(2);
    kalkulator.printTall();
}
```

Question 24
Attached



OPPGAVE 2.3

```
public class Dyr {
    public void printInfo() {
        System.out.print("Vanlig dyr, ");
    }
}

public class HonningGrevling extends Dyr {
    public void printInfo() {
        System.out.print("Honninggrevling, ");
    }
}

public static void main(String[] args) {
    Dyr dyr1 = new Dyr();
    Dyr dyr2 = new HonningGrevling();
    HonningGrevling dyr3 = new HonningGrevling();

    dyr1.printInfo();
    dyr2.printInfo();
    dyr3.printInfo();
}
```

Question 25
Attached



OPPGAVE 2.4

```
public static void main(String[] args) {  
    List<Integer> tallListe = new ArrayList<>();  
    tallListe.add(2);  
    tallListe.add(5);  
    tallListe.add(7);  
    tallListe.add(3);  
  
    Collections.sort(tallListe);  
    System.out.println(tallListe);  
}
```

Question 33
Attached



AVSPILLINGSCONTROLLER

```
public class AvspillingsController {
    @FXML
    private Label statusTekst;
    @FXML
    private ListView<Spillbar> mediaListe;
    @FXML
    private Slider progressSlider;

    public void initialize() {
        progressSlider.setShowTickMarks(true);
        progressSlider.setShowTickLabels(true);
        progressSlider.setMin(0);

        // Oppgave 3.7

        // Legger til en listener, blir kalt ved valg i listen
        mediaListe.getSelectionModel()
            .selectedItemProperty()
            .addListener(new MediaListener());
    }

    private class MediaListener implements ChangeListener<Spillbar>{
        // Metoden er forenklet noe for å gjøres oversiktlig
        @Override
        public void changed(Spillbar gammel, Spillbar ny) {

        }
    }
}
```

Question 34
Attached



JAVADOC – SLIDER

```
public class Slider  
extends Control
```

The Slider Control is used to display a continuous or discrete range of valid numeric choices and allows the user to interact with the control. It is typically represented visually as having a "track" and a "knob" or "thumb" which is dragged within the track. The Slider can optionally show tick marks and labels indicating the different slider position values.

The three fundamental variables of the slider are `min`, `max`, and `value`. The `value` should always be a number within the range defined by `min` and `max`. `min` should always be less than or equal to `max` (although a slider whose `min` and `max` are equal is a degenerate case that makes no sense). `min` defaults to 0, whereas `max` defaults to 100.

SETPROPERTY

```
public final void setMajorTickUnit(double value)  
Sets the value of the property majorTickUnit.
```

Property description:

The unit distance between major tick marks. For example, if the `min` is 0 and the `max` is 100 and the `majorTickUnit` is 25, then there would be 5 tick marks: one at position 0, one at position 25, one at position 50, one at position 75, and a final one at position 100.

This value should be positive and should be a value less than the span. Out of range values are essentially the same as disabling tick marks.

SETVALUE

```
public final void setValue(double value)  
Sets the value of the property value.
```

Property description:

The current value represented by this Slider. This value must always be between `min` and `max`, inclusive. If it is ever out of bounds either due to `min` or `max` changing or due to itself being changed, then it will be clamped to always remain valid.

SETMIN

```
public final void setMin(double value)
```

Sets the value of the property `min`.

Property description:

The minimum value represented by this Slider. This must be a value less than `max`.

SETMAX

```
public final void setMax(double value)
```

Sets the value of the property `max`.

Property description:

The maximum value represented by this Slider. This must be a value greater than `min`.

SHOWTICKLABELS

```
public final BooleanProperty showTickLabelsProperty
```

Indicates that the labels for tick marks should be shown. Typically a `Skin` implementation will only show labels if `showTickMarks` is also true.

See Also:

```
isShowTickLabels(), setShowTickLabels(boolean)
```

SHOWTICKMARKS

```
public final BooleanProperty showTickMarksProperty
```

Specifies whether the `Skin` implementation should show tick marks.

See Also:

```
isShowTickMarks(), setShowTickMarks(boolean)
```


AVSPILLINGSCONTROLLER

```
public class AvspillingsController {
    @FXML
    private Label statusTekst;
    @FXML
    private ListView<Spillbar> mediaListe;
    @FXML
    private Slider progressSlider;

    public void initialize() {
        progressSlider.setShowTickMarks(true);
        progressSlider.setShowTickLabels(true);
        progressSlider.setMin(0);

        // Legger til en listener, blir kalt ved valg i listen
        mediaListe.getSelectionModel()
            .selectedItemProperty()
            .addListener(new MediaListener());
    }

    private class MediaListener implements ChangeListener<Spillbar>{
        // Metoden er forenklet noe for å gjøres oversiktlig
        @Override
        public void changed(Spillbar gammel, Spillbar ny) {

            // Oppgave 3.8

        }
    }
}
```