Informasjon om eksamen

Oppgave	Tittel	Maks poeng	Oppgavetype
i	Egenerklæring/ Declaration, INF101		Informasjon eller ressurser
i	Generell info om digital campus eksamen - INF101, vår 22		Informasjon eller ressurser
i	Innleveringsinformasjon		Informasjon eller ressurser

Flervalg

Oppgave	Tittel	Maks poeng	Oppgavetype
1	Konstruktør	1	Flervalg
2	Person.java	4	Paring

Forklaring

Oppgave	Tittel	Maks poeng	Oppgavetype
3	Likhet	10	Langsvar
4	Book	10	Langsvar

Koding

Oppgave	Tittel	Maks poeng	Oppgavetype
5	Kortbunke	10	Langsvar
6	Whac-a-mole	35	Langsvar
7	Lever kodeoppgaver	0	Filopplasting
8	Poeng fra semesteroppgavene	30	Muntlig

Egenerklæring/ Declaration, INF101

Egenerklæring
Jeg erklærer herved at besvarelsen som jeg leverer er mitt eget arbeid.
Jeg har ikke:

- samarbeidet med andre studenter
- brukt andres arbeid uten at dette er oppgitt
- brukt eget tidligere arbeid (innleveringer/ eksamenssvar) uten at dette er oppgitt

Om jeg har benyttet litteratur *ut over pensum*, vil en litteraturliste inneholde alle kilder jeg har brukt i besvarelsen og referanser vil vise til denne listen.

Jeg er kjent med at brudd på disse bestemmelsene er å betrakte som fusk og kan føre til annullert eksamen og/eller utestengelse.

Dersom du er usikker på om du kan stille deg bak erklæringen, se **retningslinjer for bruk av kilder i skriftlige arbeider ved Universitetet i Bergen**.

Alle eksamensbesvarelser ved UiB blir sendt til manuell og elektronisk plagiatkontroll.

Merk: Ved å fortsette bekrefter jeg at jeg har lest erklæringen og at besvarelsen jeg leverer under denne eksamenen er mitt eget arbeid (og bare mitt eget arbeid), i full overensstemmelse med ovennevnte erklæringen.

Generell info om digital campus eksamen - INF101, vår 22

Eksamensinformasjon

- Eksamen inneholder 6 oppgaver
- Kandidaten må besvare alle spørsmål for å kunne oppnå full poengsum på eksamen. I tillegg vil resultatene fra semesteroppgavene legges til i endelig poengsum.
- Eksamen er delt opp i tre seksjoner beskrevet på neste side

Ta kontakt med en eksamensvakt dersom du trenger hjelp til å finne kandidatnummer eller oppgavekode.

Teknisk Hjelp
Under eksamen vil en teknisk vakt være tilgjengelig for hjelp med kodeeditor, Gitlab, hvordan zip'e en mappe, etc.

Kommunikasjon under eksamen
Under eksamen vil dere ha tilgang til internett og alle filer og programmer dere har på laptopene deres. Dere vil ha muligheten til å benytte ressurser som StackOverflow, Quora, Google, Wikipedia, etc. *Det er IKKE tillatt å kommunisere direkte med andre personer under eksamen.* Å sende meldinger på Discord, Facebook, Instagram og andre meldingstjenester, eller stille spørsmål på forum som StackOverflow og Chegg, vil ansees som **juks**.

Innleveringsinformasjon

Denne eksamenen har tre seksjoner:

1. Flervalg (oppgave 1 og 2)
2. Forklaring (oppgave 3 og 4)
3. Koding (oppgave 5 og 6)

For seksjon 3 ligger det starter-kode og instruksjoner her: https://git.app.uib.no/ii/inf101/22v/exam

Oppgavene i seksjon 1 og 2 (oppgave 1-4) skal besvares i Inspera. Vi har beregnet i underkant av 1 time til disse oppgavene. Disse oppgavene kan besvares uten å se på starter-koden fra GitLab, men all koden som vises i spørsmålene er også tilgjengelig i starter-koden om du ønsker å undersøke den nærmere.

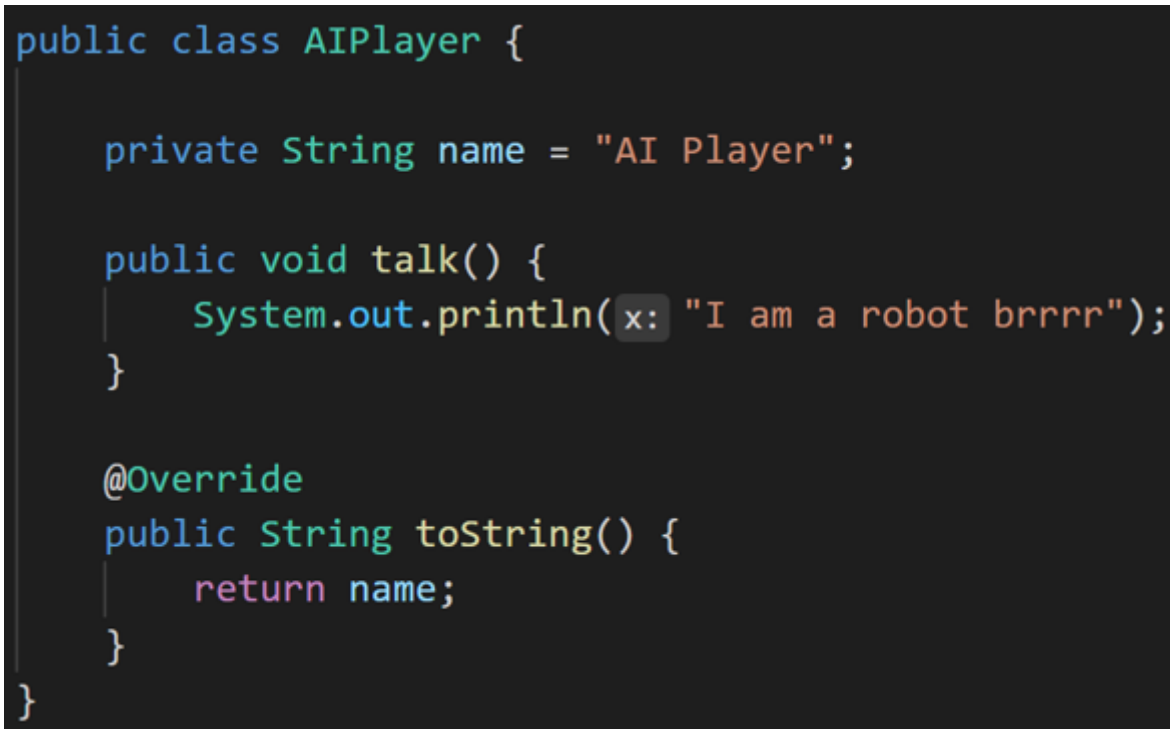
I seksjon 3 skal dere implementere funksjonalitet i et eksisterende kodeprosjekt, likt som lab- og semesteroppgavene. Vi har beregenet i overkant av 4 timer til disse oppgavene, hvorav en stor majoritet av tiden vil gå med til å utvikle en komplett GUI-applikasjon i oppgave 6.

Når dere er ferdig med kodingen skal prosjektmappen gjøres om til en zip-fil og lastes opp på Inspera. Alle kodeoppgavene skal dermed leveres som én zip-fil. Prosjektmappen som skal zippes er den mappen som inneholder blant annet *pom.xml og README.md.*

Viktig: Filen må være zippet som en *zip* -fil. Å bruke andre formater, som for eksempel *.tar.gz, .s7z, .rar, .war* eller *.jar* vil gjøre at **du mister alle poengene på kodeoppgavene!**

1 Konstruktør

Når man oppretter et nytt objekt av typen $AIPlayer$ hvor mange parametere forventer konstruktøren?



```
public class AIPlayer {

    private String name = "AI Player";

    public void talk() {
        System.out.println( x: "I am a robot brrrr");
    }

    @Override
    public String toString() {
        return name;
    }
}
```

Velg ett alternativ:

- 1
- 0
- 2
- 3

Maks poeng: 1

2 Person.java

```java
public class Person {

    private String name;
    private int yearOfBirth;

    public Person(String name, int yearOfBirth) {
        this.name = name;
        this.yearOfBirth = yearOfBirth;
    }

    public String getName() {
        return this.name;
    }

    public int getYearOfBirth() {
        return this.yearOfBirth;
    }

}
```

Hvilke av følgende konsepter brukes i klassen *Person*?

	Benyttes	Benyttes ikke
Abstraksjon	○	○
Feltvariabler	○	○
Uforanderlighet (immutability)	○	○
Polymorfisme	○	○
Instansmetoder	○	○
Innkapsling	○	○
Generiske typer	○	○
Komposisjon	○	○

Koden over er også tilgjengelig i git-repositoriet tilhørende eksamen.

Maks poeng: 4

3 Likhet

```
1  public class Main {
2
3      public static void main(String args[]) {
4          Person x = new Person("Tor", 1999);
5          Person y = new Person("To" + "r", 1999);
6          System.out.println(x == y);
7      }
8
9  }
```

Når koden over blir kjørt, printes "false" i konsollen, selv om den som skrev koden hadde forventet å få "true" fordi objektene tross alt er like. Person-klassen er den samme som i forrige spørsmål. Forklar:

- hvorfor svaret blir false, og
- hva som må gjøres for å sammenligne objekter på en god måte, og
- beskriv med ord hvordan man må endre koden for å oppnå dette.

Koden er tilgjengelig også i repositoriet tilhørende eksamen.

Skriv ca. 3 avsnitt, ikke mer enn 400 ord.
Skriv ditt svar her

Maks poeng: 10

4 Book

```
public static void main(String[] args) {
    Book haroldPoter = new Book(Arrays.asList(
        new Page(text: "Why doesn't Voldemort wear glasses?"),
        new Page(text: "Nobody nose.")
    ));

    for (Page page : haroldPoter) {
        String text = page.getText();
        System.out.println(text);
    }
}
```

Kodesnutten over har en kompileringsfeil. Forklar:

- hvorfor denne feilen oppstår, og
- hva man kan gjøre for å fikse den uten å endre noe av koden i main, og
- beskriv med egne ord stegene i løsningen din.

Under ser du koden i Book.java

```
1  public class Book {
2
3      private List<Page> pages;
4
5      public Book(List<Page> pages) {
6          this.pages = pages;
7      }
8
9      public int length() {
10         return pages.size();
11     }
12
13     public Page getPage(int pageNumber) {
14         if (pageNumber >= length())
15             throw new IndexOutOfBoundsException("Index out of bounds");
16
17         return pages.get(pageNumber);
18     }
19
20 }
```

Du finner koden i Gitlab-repositoriet i pakken *inf101v22.book*, men for denne oppgaven trenger du **IKKE endre på koden.** Dette er en forklaringsoppgave, og du skal svare her under.

Repository: https://git.app.uib.no/ii/inf101/22v/exam

Skriv ca. 3 avsnitt, ikke mer enn 400 ord.

Skriv svaret ditt her...

Maks poeng: 10

5 Kortbunke

I en standard (fransk) kortstokk har hvert kort

- en av 4 mulige *"farger"* (suits): kløver, ruter, hjerter eller spar; og
- en av 13 mulige *valører* (ranks): to, tre, fire, fem, seks, sju, åtte, ni, ti, knekt, dronning, konge eller ess.

Det finnes dermed totalt 52 ulike kort. I repositoriet finner du en kort-klasse *Card* som modellerer et slikt kort.

En kortbunke er en samling med kort.

I denne oppgaven skal du implementere en kortbunke på to forskjellig måter. La begge klassene du oppretter være i pakken *inf101v22.cards*.

1. Implementer en kortbunke-klasse *InheritedArrayCardPile* ved bruk av **arv** fra $ArrayList$.
2. Implementer en kortbunke-klasse *ComposedArrayCardPile* ved bruk av **komposisjon** av $ArrayList$.

For begge klassene skal du:

- Implementere en metode $createFullDeck()$ som returnerer en ny bunke med kort. Metoden skal returnere et nytt kortbunke-objekt med alle de 52 forskjellige kortene. Burde denne metoden være static eller non-static?
- Implementere en metode med signatur $boolean\ add(Card)$ som legger til et kort i bunken. Metoden skal kaste IllegalArgumentException dersom noen forsøker å legge til null, men skal ellers legge til kortet i bunken og alltid returnere true (rart, men slik er spesifikasjonene). Burde denne metoden være static eller non-static?

Man kan se for seg at en klasse som representerer en bunke med kort burde ha flere metoder for å være en fullgod modell, men i denne oppgaven trenger du kun å implementere de to nevnte metodene.

Klon Gitlab-repositoriet, skriv koden og zip prosjektmappen når du er ferdig med alle kodeoppgavene. Du laster opp zip-filen på siste spørsmål.

Repository: https://git.app.uib.no/ii/inf101/22v/exam
Eventuelle kommentarer på oppgaven kan gis i tekstfeltet nedenfor.

Maks poeng: 10

6 Whac-a-mole

I tivoli-spillet whac-a-mole er poenget å banke ned så mange mulvarper som mulig i løpet av en gitt tid. Hver gang en mulvarp blir banket, dukker det opp en ny mulvarp et tilfeldig sted, som så skal bankes ned. Dersom man banker ned veldig mange mulvarper, kan man vinne en premie.

I denne oppgaven skal du lage en digital versjon av whac-a-mole. Du bør følge instruksjonene så tett som mulig slik at du demonstrerer at du forstår hva som menes.

Instruksjonene og koden finner du i Gitlab-repositoriet.

Klon Gitlab-repositoriet, skriv koden og zip prosjektmappen når du er ferdig med alle kodeoppgavene. Du laster opp zip-filen på siste spørsmål.

Repository: https://git.app.uib.no/ii/inf101/22v/exam

Eventuelle kommentarer på oppgaven kan gis i tekstfeltet nedenfor.

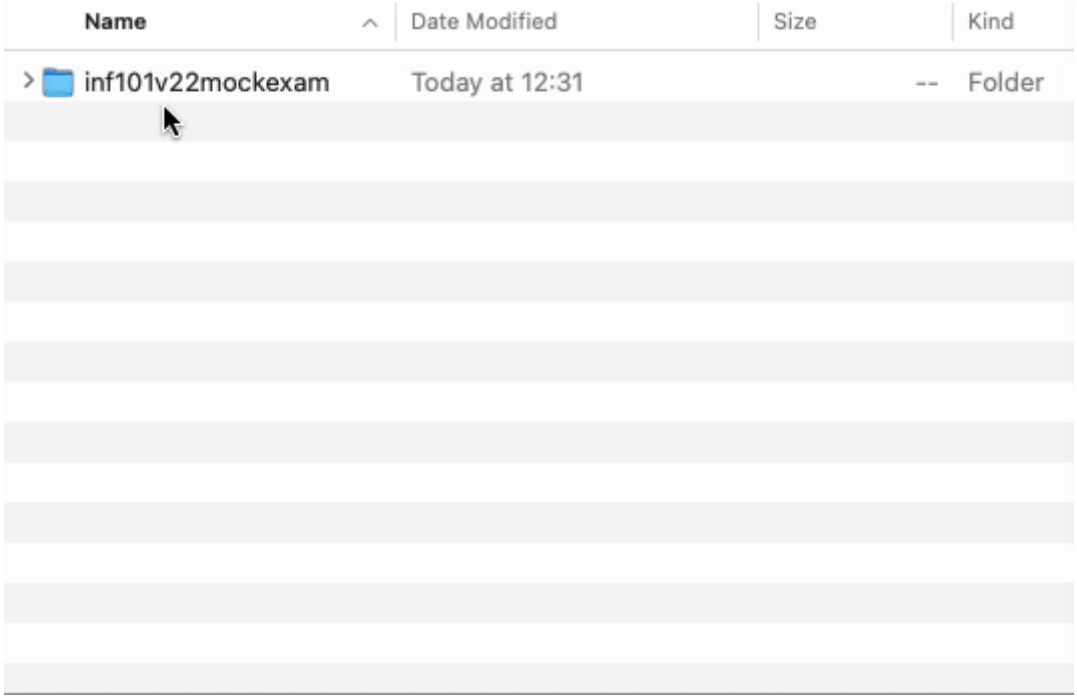
Maks poeng: 35

7 Lever kodeoppgaver

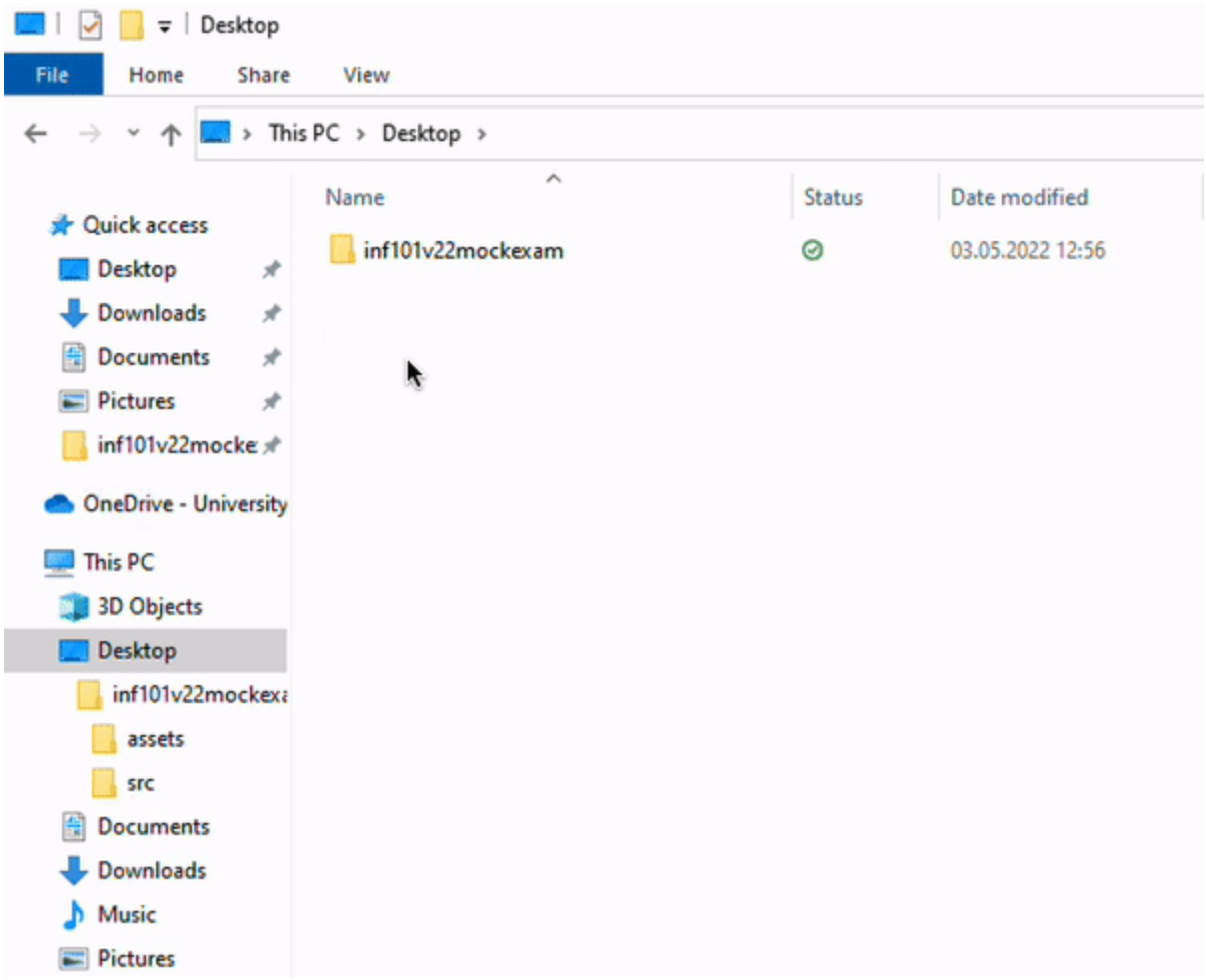
Last opp prosjektmappen som en zip-fil. Det er mappen som inneholder *pom.xml.*

Viktig: Mappen må være zippet som en *zip* -fil. Å bruke andre formater, som for eksempel *.tar.gz, .7z, .rar, .war* eller *.jar* vil gjøre at **du mister alle poengene på kodeoppgavene!**

Hvordan lage .zip på Mac
- Høyreklikk riktig mappe (altså mappen som inneholder pom.xml)
- Velg compress



Hvordan lage .zip på Windows
- Høyreklikk riktig mappe (altså mappen som inneholder pom.xml)
- Velg send to -> compressed (zipped) folder



Hvordan lage .zip på Ubuntu
- Høyreklikk riktig mappe (altså mappen som inneholder pom.xml)
- Velg compress
- I menyen som dukker opp, velg .zip

Maks poeng: 0

8 Poeng fra semesteroppgavene

Dine poeng fra semesteroppgavene vil legges til her. Det er ingen ting du kan gjøre fra eller til på dette punktet nå.

Maks poeng: 30