

Tetseksamen

02100

Indledende Programmering og Softwareteknologi, Efterår 2025

Student: s255955**Due:** 03 Dec 2025

Programming Language: Java

Opgave 1

```
public static void figure(int n){
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < i; j++) {
            System.out.print(" ");
        }
        for(int j = 0; j < (2 * n) - 1 - (i*2); j++){
            System.out.print("*");
        }
        System.out.println();
    }
}
```

Opgave 2

```
import java.util.Scanner;
import java.util.ArrayList;

public class Loop {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);
        ArrayList<String> words = new ArrayList<>();
        System.out.println("Please enter \"STOP\" to stop.");
        while (true) {
            System.out.print("Enter a word: ");
            String next_word = console.next();
            words.add(next_word);
            if (next_word.equals("STOP")) {
                break;
            }
        }
        System.out.println("Here are the word lengths:");
        for (int i = 0; i < words.size(); i++) {
            String word = words.get(i);
            System.out.println(word + " " + word.length());
        }
        System.out.println("The last word was \"STOP\" of course.");
    }
}
```

```
}  
}
```

Opgave 3

```
public class Data {  
    private int major;  
    private int minor;  
    private boolean active;  
    public Data(int major, int minor, boolean active) {  
        this.major = Math.max(0, major);  
        this.minor = Math.max(0, minor);  
        this.active = active;  
    }  
  
    public int getMajor() {  
        return this.major;  
    }  
    public int getMinor() {  
        return this.minor;  
    }  
    public boolean getActive() {  
        return this.active;  
    }  
    public void parity() {  
        this.major = this.major % 2;  
        this.minor = this.minor % 2;  
    }  
    public String toString() {  
        return "" + this.major + (this.active ? "***" : "/") + this.minor;  
    }  
}  
  
public class Grid {  
    private Data[] a;  
    public Grid(boolean[][] m) {  
        this.a = new Data[m[0].length * m.length];  
        for (int i = 0; i < m.length; i++) {  
            for (int j = 0; j < m[i].length; j++) {  
                this.a[(m[0].length * i) + j] = new Data(i, j, m[i][j]);  
            }  
        }  
    }  
    public void update() {  
        for (int i = 0; i < this.a.length; i++) {  
            if (this.a[i].getActive()) {  
                this.a[i].parity();  
            }  
        }  
    }  
}  
  
    public String toString() {  
        String return_string = "";  
        for (int i = 0; i < this.a.length; i++) {  
            return_string += this.a[i] + (i == this.a.length - 1 ? "" : "-");  
        }  
    }  
}
```

```
    }  
    return return_string;  
}  
  
}
```

Opgave 4

```
public class Person {  
    private String navn;  
    public Person(String navn) {  
        this.navn = navn;  
    }  
  
    public String toString() {  
        return this.navn;  
    }  
}
```

```
public class Vektor extends Person {  
    private int id;  
    public Vektor(String navn, int id) {  
        super(navn);  
        this.id = id;  
    }  
    public String toString() {  
        return "Vektor:" + super.toString() + ";" + this.id;  
    }  
}
```

```
public class Rektor extends Person {  
    private double id;  
    public Rektor(String navn, double id) {  
        super(navn);  
        this.id = id;  
    }  
    public String toString() {  
        return "Rektor:" + super.toString() + ";" + this.id;  
    }  
}
```

```
public class Lektor extends Person {  
    public Lektor(String navn) {  
        super(navn);  
    }  
    public String toString() {  
        return "Lektor:" + super.toString();  
    }  
}
```