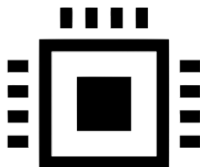


# Arduinoprogrammering

## Øving 2

IELET1002



DATATEK.

DIGITAL KOPI

## Bakgrunn:

Denne øvingen skal dere jobbe med i prosjektgruppene, men besvarelsen leveres individuelt.

Målsettingen med øvingen er som følger:

- Bli vant til å jobbe sammen i gruppene, og lære dere å sammen gå framover når dere skal løse en oppgave dere blir presentert for (forberedelse til prosjektet med andre ord).
- Komme i gang med å bruke Arduino-kit'ene.
- Oppgavene i øvingen diskuteres én og én i gruppa (dvs. framgangsmåte for å finne en løsning), og så løser den enkelte oppgaven på egen hånd og leverer på vanlig måte i Blackboard. *Vi ønsker å unngå at de som ikke umiddelbart forstår å løse en oppgave, kopierer løsningen fra andre - for da har gruppen ikke bestått "testen".*
- Det handler altså mye om **kompetanseoverføring**: de som kan tematikken som ligger i en oppgave, tar ansvar og sørger for at alle i gruppa forstår. Det er faktisk en vinn-vinn-situasjon, hvor både den som lærer bort og den som blir hjulpet, forstår stoffet bedre etterpå.
- Gruppene finner selv ut hvordan de skal gjøre dette rent praktisk, dvs. gjennomføringen av kompetanseoverføringen, og avsjekken på at alle har forstått og fått løst oppgavene i øvingen.
- Det skal skrives et refleksjonsnotat (se egen deloppgave i øvingen) som både tar for seg hvordan gruppa som helhet ble enige om å gjennomføre øvingen og kompetanse-overføringen, og hvordan det enkelte gruppe medlem opplevde at det fungerte.

## Forberedelser til øvingen:

Det er lagt ut en del relevant støttemateriale på Blackboard, og jeg har markert dette med rød pil i utklippet under.

**Støttemateriell**

**Arduino videoressurser**

Hva er en mikrokontroller?: <https://youtu.be/jcrFvTiders>

Arduino Starter Kit: <https://youtu.be/LzuYMHQcQrl>

Arduino Software IDE: <https://youtu.be/5qNUwfvXAmE>

TinkerCAD 1: <https://youtu.be/XigRvKLGaAo>

TinkerCAD 2: <https://youtu.be/dZoY-vwgrk>

DigitalWrite og pinMode: <https://youtu.be/AINnpsbDl>

Funksjoner 1: <https://youtu.be/NjDEWLS2XM4>

Funksjoner 2: <https://youtu.be/XTJoB-hkBu4>

Switch Case med oppgave: <https://youtu.be/fbzWCNFZsU>

Switch Case med løsning: <https://youtu.be/kOknxHnHqQA>

**Hjelperessurs 1-3**

Vedlagte filer:

- Hjelperessurs 01 - IDE og Ditt Første Program (546,119 kB)
- Hjelperessurs 02 - Variabler og Datatyper (1 018,608 kB)
- Hjelperessurs 03 - Trykknappkretser (1,076 MB)

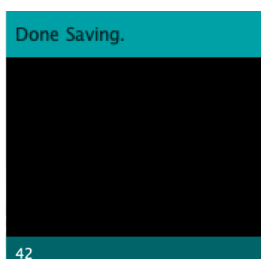
- Hjelperessurs 1: IDE og ditt første program  
- Hjelperessurs 2: Variabler og Datatyper  
- Hjelperessurs 3: Trykknappkretser

## Oppgave 1 - Bli bedre kjent med Arduino IDE

1. Hva står IDE for?
2. Hva betyr §-tegnet ved siden av programnavnet i Arduino IDE?



- a) Programmet har endringer som ikke er lagret
  - b) Arduino-kortet er ikke tilkoblet
  - c) Det er feil i programmet
3. Når du trykker på Verify-knappen så lastes programmet også opp til Arduino-kortet
    - a) True
    - b) False
    - c) Det kommer an på
  4. Når du trykker på Upload-knappen så verifiseres koden også etterpå
    - a) Det kommer an på
    - b) Sann
    - c) Usann
  5. Når du oppretter en ny tab i Arduino IDE, så etableres en separat
    - a) Folder
    - b) Folder og fil
    - c) Fil
  6. Tallet nederst til venstre i Arduino IDE betyr



- a) Kodelinjen som kursoren er på akkurat nå
- b) Antall funksjoner som programmet kaller, slik at du kan estimere minnebruken
- c) Hvor mange kodelinjer det er i programmet

7. Arduino IDE kan støtte mange forskjellige typer kort (det vanligste er Arduino/Genuino UNO).  
Verktøyet som benyttes for å utvide støtten til andre kort kalles (Google denne)
  - a) Boards Manager
  - b) Bootloader
  - c) Library Manager
8. Tastaturnarveien for Verify er (Mac: bytt ut CTRL med CMD)
  - a) SHIFT + CTRL + U
  - b) CTRL + T
  - c) CTRL + R
9. Tastaturnarveien for å åpne Serial Monitor er (Mac: bytt ut CTRL med CMD)
  - a) CTRL + T
  - b) CTRL + R
  - c) SHIFT + CTRL + M
10. Det er en knapp for å kopiere feilmeldinger i Arduino IDE
  - a) True
  - b) False
11. I Arduino IDE Preferences kan du justere alt i lista nedenfor, unntatt...
  - a) Endre fontstørrelsen i editoren
  - b) Sjekke om det er nye versjoner av Arduino IDE ved oppstart
  - c) Endre lagringslokasjonen for programmene dine
  - d) Slå av feilmeldinger
12. Når du har skrevet et Arduino nøkkelord korrekt vil Arduino IDE endre nøkkelordets
  - a) Størrelse
  - b) Font
  - c) Farge
  - d) Innrykk

## Oppgave 2 - analogRead() og spenningsdeling

a)

**Kobling:** Potensiometer (potmeter) er en 3-pinnars variabel resistor, og kan alene benyttes som spenningsdeler - dvs. samme kobling som med lyssensor og fastmotstand fra tidligere eksempel vi har sett på, men både lyssensoren og fastmotstanden er erstattet med potmeteret. Google hvordan et potmeter fungerer dersom du er usikker.

Merk! Midtpinnen på potmeteret skal aldri kobles til jord (GND) eller +5V, for da er sjansen stor for at du lager kortslutning mellom dem = ødelagt Arduino!

**Program:** Når du dreier på potmeteret endres verdien inn på den analogpinne du har koblet til midtpunktet. Mens du sakte dreier potmeteret fra den ene ytterposisjonen til den andre skal programmet kontinuerlig lese verdien på den aktuelle analogpinne, og skrive den til Serial Monitor.

**Krav:** Arduino-kit. Lever både kommentert kode, observasjon/diskusjon og et koblingsskjema som du tegner (håndtegnet og limt inn i besvarelsen er OK).

**Observer og diskuter i besvarelsen:** Hvilket verdispenn ser du i Serial Monitor (dvs. minimum til maksimum)? Stemmer dette med "teorien"? Dersom du plugger ut potmeter-ledningene som er koblet til Arduino-kortet, og i stedet gjør følgende (mens programmet fortsatt kjører, men vær forsiktig så du ikke lar noen av de løse ledningene berøre hverandre):

i) plugg den ene enden av en ledning i analogpinne og den andre i en jordpinne (GND):  
- Hvilken verdi leser du i Serial Monitor?

ii) trekk ut den enden som er koblet til jord, og sett den inn i +5V i stedet:  
- Hvilken verdi leser du nå?

Der skal du ha fasiten på hva verdispennet bør være når du skrur potmeteret fra ytterpunkt til ytterpunkt. Fikk du samme verdispenn i begge tilfellene? Hvis nei, forklar hvorfor.

b)

**Kobling:** Som i a)

**Program:** Modifiser programmet i a) slik at Serial Monitor i stedet skriver ut spenningsverdien lest på analogpinne

**Krav:** Arduino-kit. Lever kommentert kode og observasjon/diskusjon.

**Observer og diskuter i besvarelsen:** Forklar hvordan du fant spenningsverdiene. Hvilken spenning er det egentlig snakk om her, hvilket spenningsfall er det vi måler?

c)

**Kobling:** Som i a)

**Program:** Modifiser programmet i b) ytterligere slik at det beregner den motstandsverdien du måler spenningsfallet over, og skriver ut både U og tilhørende R i Serial Monitor.

**Krav:** Arduino-kit. Lever kommentert kode og observasjon/diskusjon.

**Observer og diskuter i besvarelsen:** Forklar hvordan du tenkte da du fant R.

d)

**Kobling:** Som i a)

**Program:** Helt nytt program, ny funksjonalitet. Du leser riktignok fortsatt verdiene på analogpinne (ikke omregnet til spenning) mens du sakte dreier potmeteret fra ytterpunkt til ytterpunkt, men nå skal du kalle

en funksjon med de avleste verdiene. Funksjonen skal så mappe mottatte verdier fra det avleste verdispenn til heltallsspennet 0-3, og returnere 0, 1, 2, eller 3. Void loop() tar deretter den returnerte verdien og skriver "range x" til Serial Monitor, hvor x = 0, 1, 2, eller 3 avhengig av hvor funksjonen mappet den mottatte verdi. Legg gjerne inn en liten delay på 100ms el.lign. mellom hver gang funksjonen kalles.

**Krav:** Arduino-kit. Lever kommentert kode og bilde av kretsen.

### Oppgave 3 - analogWrite og PWM

<https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/>

a)

**Kobling:** En LED-krets tilkoblet en PWM-pinne på digitalporten

**Program:** Lysstyrken i LEDen skal gradvis blir sterkere til den er maksimal, for så å bli gradvis svakere til den er null. Dette gjentas om og om igjen. Det skal se ut som om LEDen "puster".

**Krav:** Arduino-kit'et

b)

**Kobling:** Lyssensorkrets koblet til en analogpinne. En LED-krets koblet til en PWM-pinne på digitalporten

**Program:** Lysmengden som måles med lyssensoren skal styre lysstyrken i LEDen. Jo høyere verdi fra lyssensoren, desto mindre lysstyrke i LEDen.

**Krav:** Arduino-kit'et. "Juster" koden på en sånn måte at når du klemmer tre fingre ganske hardt rundt hodet på sensoren, og ikke noe lys slipper inn, så skal LEDen lyse sterkest.

### Oppgave 4 - Random

a)

**Kobling:** Ingen eksterne kretser, bare Arduinokortet

**Program:** Et tilfeldig tall mellom 1 og 6 skal genereres hvert sekund og skrives til Serial Monitor.

**Krav:** Arduino-kit'et. Bruk en åpen pinne på analogporten til å generere såkornet (seed number) til random-funksjonen. Du får ikke lov til å bruke delay().

**Hint:** <https://www.arduino.cc/reference/en/language/functions/random-numbers/random/>

b)

**Kobling:** En buzzer med en seriemotstand på 1kΩ - tilknyttet mellom jord og en digitalpinne

**Program:** Som i a), men hver gang tallet 6 genereres skal det komme et kort pip (f.eks. 1kHz i 200ms)

**Krav:** Som i a)

**Hint:** Sjekk ut instruksjonen `tone()` i Arduino Language Reference på [arduino.cc](http://arduino.cc)

c)

**Kobling:** Ingen eksterne kretser, bare Arduinokortet

**Program:** Et tilfeldig tall mellom 1 og 6 skal genereres uten "delay"-element 10.000 ganger, og det skal skrives en rapport til Serial Monitor: Seks linjer, én for hvert tall (her eksemplifisert ved linje 1):

Tallet 1 ble generert "x"-ganger og det tilsvarer "y" prosent

Deretter må du vente litt før `void loop()` får kjøre samme test igjen.

**Krav:** Arduino-kit'et. Bruk en åpen pinne på analogporten til å generere såkornet (seed number) til random-funksjonen. Utskriften til Serial Monitor skal løses med en for-løkke som kjører 6 ganger.

d)

**Kobling:** En trykknappkrets

**Program:** Når trykknappen trykkes, skal et tilfeldig tall mellom 1 og 6 genereres uten "delay"-element 10.000 ganger. Trykkes knappen en gang til, skal det skrives en rapport til Serial Monitor: Seks linjer, én for hvert tall (her eksemplifisert ved linje 1):

Tallet 1 ble generert "x"-ganger og det tilsvarer "y" prosent

Deretter må du vente litt før `void loop()` får kjøre samme test igjen.

**Krav:** Arduino-kit'et. Bruk en åpen pinne på analogporten til å generere såkornet (seed number) til random-funksjonen. Dersom knappen trykkes på nytt, før det har blitt kjørt en 10.000-test, skal det skrives bare en melding til Serial Monitor: "No test data available yet".

## Oppgave 5 - Refleksjonsnotat

Her skal hvert enkelt gruppemedlem reflektere over kompetanseoverføringselementet og gruppesamarbeidet som beskrevet i innledningen til øvingen.

a)

Reflekter over den planen som ble lagt av gruppen for å sikre det påkrevde elementet av kompetanseoverføring i øvingen. Hvordan oppfattet du at det skulle gjennomføres, og hva var dine tanker om det?

b)

Reflekter over hvordan kompetanseoverføringen var i praksis

c)

Hva bør gruppen endre, øve mer på, forsterke, etc. når gruppa skal samarbeide i prosjektet?