

# Innebygde datasystemer eksamen 2020

## Oppgave 1

- C++ er et typet språk. Dette vil si at:
  - ☐ Påstanden er usann
  - ☐ En variabel kan deklarerer med auto, og har da ikke en datatype
  - ☐ Det er ulovlig å konvertere mellom datatyper
  - ☐ En variabel har alltid en datatype

## Oppgave 2

Hva slags datatype er dette?

```
1 enum class Operation { predict=1, update=0 };
```

- Velg et alternativ:
  - ☐ En enum med to mulige verdier, predict og update
  - ☐ Klasse, med to medlemsvariable
  - ☐ Dette er ikke en datatype
  - ☐ int (heltall)

## Oppgave 3

Vi ønsker å deklarere et heltall uten fortegn, og med fast bredde 16 bit. Hvilken datatype velger du? Det kan antas at C++11 eller nyere er tilgjengelig.

Svar her:

## Oppgave 4

Programmet ditt skal utføre en gjentakende oppgave 34 ganger. Antallet ønskede iterasjoner er altså kjent på forhånd. Hvilken type løkke er mest hensiktsmessig for dette formålet?

- Velg ett alternativ:

- ☐ do/while
- ☐ while
- ☐ loop
- ☐ for

## Oppgave 5

Klassen KryptoWallet inneholder én medlemsvariabel og én public funksjon. Funksjonen skal returnere konto-ID, og medlemsvariabelen inneholder den samme konto-ID. Hvilket navn/identifikator er mest korrekt å velge til hver av dem? (Kryss av i tabell)

	accountId_	accountId	_accountId_	Account_ID
Medlemsfunksjon				
Medlemsvariabel				

Table 1: Caption

## Oppgave 6

Programmet skal skrive ut MAS234! til terminal. Tegnet backslash n på slutten gir linjeskift. Fyll inn det som mangler for at dette enkle programmet skal kompilere og kjøre! Husk å kvalifisere navnet.

```
#include <iostream>

int main()
{
     << "MAS234!\n";
    return 0;
}
```

## Oppgave 7

Klasse og x (enum, class enum, struct) er nesten det samme i C++. Den eneste signifikante forskjellen er default synlighet på funksjoner og variable. (Hva skal stå for x?)

## Oppgave 8

Synlighet for variabler og funksjoner tilhørende en klasse styres med public, private og/eller protected. Hva gjør private?

Ett av svarene under gir 2 poeng, ett av svarene gir 0 poeng, og ett av svarene gir -1 poeng.

- Velg ett alternativ:
  - ☐ Private øker sikkerheten, og forhindrer 100% at noen kan hente ut data fra minnet.
  - ☐ Funksjoner og variable deklart private kan kun aksesseres fra andre funksjoner i klassen selv.
  - ☐ Variabler deklart private kan alltid leses fra utsiden av klassen på samme måte som private-variabler

## Oppgave 9

Hva er den største verdien som kan legges inn i en 16 bit unsigned int?

Svar:

## Oppgave 10

C++ er standardisert av ISO. Hvorfor er det viktig at språket er standardisert? Det finnes mange gode grunner til å standardisere, men kun ett alternativene her er korrekt.

- Velg ett alternativ:
  - ☐ Programmer skrevet i standard C++ skal fungere med kompilatorer fra ulike leverandører
  - ☐ Det blir totalt sett billigere å ta frem en ny versjon av språket når mange deltar
  - ☐ Ingen deler av språket utvikler seg videre
  - ☐ Det går raskere å ta frem en ny versjon av språket

## Oppgave 11

I denne oppgaven ønsker vi å returnere to int-verdier fra en C++-funksjon. Hvilken funksjons-deklarasjon er korrekt for dette formålet?

- Velg ett alternativ:
  - ☐ void calculate(int& v1, int & v2, const int input);
  - ☐ int, int calculate(const int input);
  - ☐ double calculate(const int input);
  - ☐ int&, int& calculate(const int input);

## Oppgave 12

```

#ifndef MYHEADER_H
#define MYHEADER_H

#include <cmath>

double unknownName(double a, double b) {
    return 1.0 + std::pow(a, b);
}

#endif // MYHEADER_H

```

Hva beregner funksjonen unknownName i illustrasjonen? Ditt svar her:

## Oppgave 13

```

enum class Color { red, black };

bool verifyPercentage(std::vector<Color>& nonstop)
{
    int r = 0;
    int b = 0;
    for(auto& color : nonstop) {
        switch (color)
        {
            case Color::black:
                ++b;
                break;
            case Color::red:
                ++r;
                break;
        }
    }

    return r == b;
}

```

- Velg ett alternativ:
  - ☐ Alt er perfekt, kjør på!
  - ☐ enum class finnes ikke!
  - ☐ Implementasjonen itererer ett hakk for langt, og ligger ikke i korrekt namespace.
  - ☐ Implementasjonen er korrekt, men må legges i namespace CakeControl.

## Oppgave 14

De fire tegnene som printes er: \_\_\_\_\_

```
#include <iostream>

int main()
{
    char a[] = { 'M', 'A', 'S', '2', '3', '4', 'Y' };
    char* p1 = &a[4];
    char* p2 = &a[5];

    p2++;

    std::cout << *p1 << a[1] << a[2] << *p2 << std::endl;
}
```

## Oppgave 15

- Velg ett alternativ:

- ☐ Ingen av disse.
- ☐ 1
- ☐ 1 2
- ☐ 1 2 3.

## Oppgave 16

- Velg ett alternativ:

- ☐ Programmet kompilerer ikke
- ☐ Følgende printes til terminal: Value is 43
- ☐ -.- Value is 42
- ☐ Programmet kjører, men krasjer før cout

```
#include <iostream>

int main()
{
    const char m = 42;
    const char* a = &m;
    const char* s = a;

    std::cout << "Value is: " << ++(*s) << std::endl;
    return 0;
}
```

## Oppgave 17

Hvis vi oppretter en double-variabel som IKKE skal kunne endre verdi etter opprettelsen

- Velg ett alternativ:
  - ☐ Skjule variabelen i et namespace
  - ☐ Deklarere variabelen const
  - ☐ Lage en privat variabel i en klasse
  - ☐ Deklarere variabelen static
  - ☐ Velge en enum

## Oppgave 18

- Velg ett alternativ:
  - ☐ Programmet kompilerer, men vil ikke starte
  - ☐ Programmet kompilerer ikke
  - ☐ Det kan vi ikke vite sikkert. Dette er et eksempel på udefinert oppførsel
  - ☐ De fem elementene i array-en printes ut etter hverandre

```
#include <iostream>

int main()
{
    double measuredValues[5] {1.0, 5.0, -2.0, 8.3, 5.1 };

    for (int i = 0; i <= 5; i++)
    {
        std::cout << measuredValues[i] << ", ";
    }
    return 0;
}
```

## Oppgave 19

Hva gjør følgende C++-kodesnutt? `PORTB &= ~(1U << 3U);`

- Velg ett alternativ:
  - ☐ Bit 3 på PORTB toggles, mens de andre IKKE påvirkes.
  - ☐ Bit 3 på PORTB settes til lav, mens de andre bit-ene IKKE påvirkes.
  - ☐ Dette er feil, U har ikke noe her å gjøre.
  - ☐ Bit 4 på PORTB settes til lav, mens de andre bit-ene settes høye

```
PORTB  &=  ~(1U  <<  3U) ;
```

## Oppgave 20

Hva gjør data direction-registerene på en AVR mikrokontroller?

- Velg ett alternativ:
  - ☐ Styrer intern pull-up-motstand.
  - ☐ Styrer retningen på CAN-bus.
  - ☐ Styrer retning på pinnene på en gitt port.
  - ☐ Styrer UART, slik at vi kan bruke printf.

## Oppgave 21

Alle C++-funksjoner unntatt konstruktører og destruktører må ha returtype. Hvilken returtype benyttes når vi IKKE ønsker å returnere noe?

Ditt svar her: -----

## Oppgave 22

Funksjoner i C++ kan kalle seg selv. Dette heter rekursjon. Hva er essensielt for en rekursiv funksjon?

- Velg ett alternativ:
  - ☐ Den må ikke ta inn parametre.
  - ☐ Det må legges inn en stoppbetingelse, slik at den maksimalt kaller seg selv N ganger.
  - ☐ Den må ikke ha returtype, slik at det unngås kopiering av data for hver runde

## Oppgave 23

Vi kan ha mer enn én destructor i en C++-klasse.

- Velg ett alternativ:
  - ☐ Sant
  - ☐ Usant

## Oppgave 24

Vi kan ha mer enn én konstruktør i en C++-klasse

- Velg ett alternativ:

- ☐ Sant
- ☐ Usant

## Oppgave 25

Når skal destructor i C++ deklarerer virtual?

- Velg ett alternativ:

- ☐ Hvis vi eller andre kan tenkes å arve fra eller til klassen, nå eller senere, så skal den være virtual.
- ☐ Bare hvis klassen er abstrakt.
- ☐ Det er unødvendig – også ved arv.
- ☐ Det gjør programmet hakket raskere, men er ikke nødvendig. Hvis vi har store programmer, så bør destructor deklarerer virtual i de fleste tilfeller.

## Oppgave 26

Hvor mange klasser kan vi arve fra i samme klasse i C++ ? Altså; hvor mange ”foreldreklasser” kan en klasse ha?

- Velg ett alternativ:

- ☐ Maksimalt 2.
- ☐ Ingen grense av praktisk betydning (vi kan ha over 1000 foreldre-klasser).
- ☐ Maksimalt 10.
- ☐ Maksimalt 1.

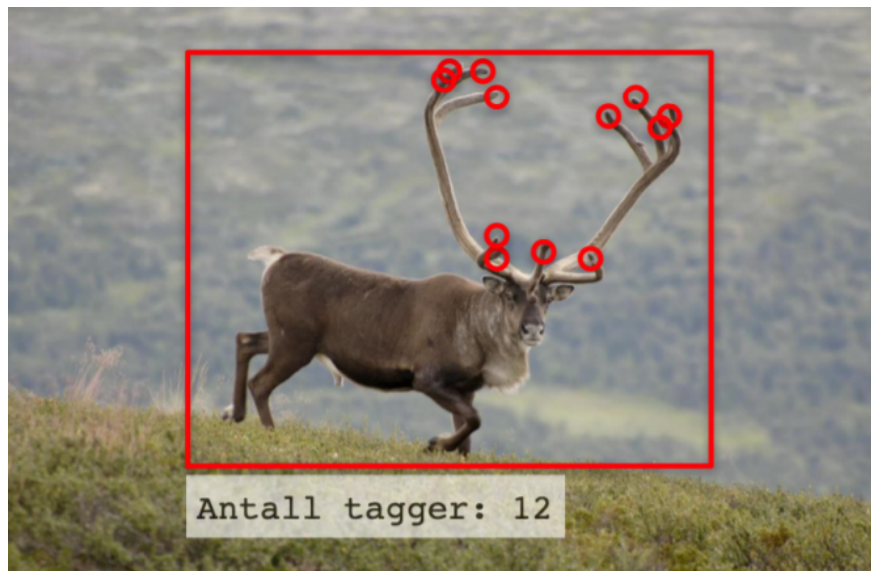
## Oppgave 27

Hvilken påstand er riktigst:

- Velg ett alternativ:

- ☐ I C++ kalles alle funksjonene i Main-filen ved oppstart.
- ☐ Main-funksjonen er alltid startpunktet i et C++-program.
- ☐ Operativsystemet kaller main-funksjonen når et C++-program startes.
- ☐ Vi kan fritt velge navn på første funksjon som kalles i et C++-program.





## Oppgave 28

Hvilket alternativ beskriver IKKE et tilfelle med behov for krysskompilering.

- Velg ett alternativ:
  - ☐ Kompilering av kildekode for iPad med ARM på en Intel x86 MS Windows-maskin.
  - ☐ Kompilering av kildekode for embedded Linux på maskinen dette skal kjøres på (f.eks. Raspberry Pi).
  - ☐ Kompilering av kildekode for mikrokontroller fra MacOS
  - ☐ Kompilering av kildekode for Arduino på AMD64 PC med MS Windows.

## Oppgave 29

Påstand: Innebygde datasystemer har lav ytelse pga. bruk av Arduino og mikrokontrollere.

- Velg ett alternativ:
  - ☐ Sant, mikrokontrollerene på Arduino er trege.
  - ☐ Usant, det finnes mange godt egnede plattformer med høy ytelse, som ikke nødvendigvis er mikrokontrollerbasert. Mikrokontrollere kan også gi høy ytelse til en rekke formål.

## Oppgave 30

Det skal utvikles styrings-computer for en drone på i 250 gram. Hva vil du vektlegge? Pris er allerede med i vurderingen.

- Velg ett alternativ:

- ☐ Høy ytelse og tilgjengelig IO.
- ☐ Lav vekt, tilgjengelig IO og tilstrekkelig ytelse.
- ☐ Lav vekt, lavt effektforbruk og høy ytelse.

## Oppgave 31

Hva har GIT versjonskontroll til felles med kryptovalutaen Bitcoin?

- Velg ett alternativ:
  - ☐ Kryptosjekksummer benyttes for å sikre at ingen kan tukle med/endre historikk uten at det detekteres.
  - ☐ Begge deler er shady.
  - ☐ Utviklere er sikret anonymitet fordi historikk enkelt kan endres i ettertid. Det er viktig hvis f.eks. en privat epost-adresse ved en feil legges inn som kommentar i kildekode som sjekkes inn.
  - ☐ Samme person står bak / har funnet opp begge deler, og GIT er avhengig av Bitcoin for å fungere.

## Oppgave 32

Hva slags type versjonskontroll er GIT?

- Velg ett alternativ:
  - ☐ En webside
  - ☐ Distribuert (potensielt)
  - ☐ Lokal (unngå konflikter)
  - ☐ Sentralisert (alltid)

## Oppgave 33

Du skal hente et GIT-repository fra Github for å gjøre noen forbedringer. Hva er korrekt fremgangsmåte fra terminalen? Anta at du står i rett folder, der du ønsker repository-et plassert. Anta også at repository free-falling-cat eksisterer. (Det eksisterer imidlertid ikke i virkeligheten, så samtlige alternativer vil feile i praksis!).

- Velg ett alternativ:
  - ☐ `git checkout https://github.com/uia-mechatronics/free-falling-cat.git`
  - ☐ `git clone https://github.com/uia-mechatronics/free-falling-cat.git`
  - ☐ `git rework https://github.com/uia-mechatronics/free-falling-cat.git`
  - ☐ `git copy uia-mechatronics:free-falling-cat.git`

## Oppgave 34

Terminalen i f.eks. Ubuntu Linux er et kraftig verktøy, og kjøring av kommandoer dere ikke vet hva gjør, må unngås. Dette er spesielt viktig dersom vi er innlogget som rotbruker (root) eller når vi kjører noe med superbruker-rettigheter (sudo). Anta at du er innlogget som bruker root. Hvilken av de følgende kommandoer er farligst? ADVARSEL: IKKE PRØV KOMMANDOENE!

- Velg ett alternativ:
  - ☐ `rm -rf /`
  - ☐ `touch /home/username/fil.txt`
  - ☐ `fdisk -l`
  - ☐ `wget http://some-random-website.net/CppClass.cpp`

## Oppgave 35

For å sjekke at overføring av data, f.eks. over en serielinje, har gått bra, kan vi benytte feildeteksjon. Sortér de følgende typene feildeteksjon slik at nummer 1 er den med mulighet til på detektore flest typer overføringsfeil, og nummer 3 er den som detekterer færrest typer feil.

	2	1	3
Sjekksum (f.eks SHA256)			
Cyclic Redundancy Check (CRC)			
Paritetsbit			

Table 2: Caption

## Oppgave 36

Hva er en nettverkstopologi?

- Velg ett alternativ:
  - ☐ En nettverkstopologi gir oss adressene på enhetene i nettverket.
  - ☐ En nettverkstopologi beskriver hvordan noder i nettverket er koblet sammen.
  - ☐ En nettverkstopologi forteller hvilken type nettverk som benyttes (f.eks. CAN og Ethernet).
  - ☐ En nettverkstopologi beskriver hvordan noder i nettverket er koblet sammen, for trådløse nettverk har vi derfor ingen topologi.

## Oppgave 37

Du skal sjekke uvanlige feilmeldinger på en datamaskin som overvåker bestanden av reinsdyr på Hardangervidda. Maskinen er tilkoblet internett med public IP, og du har satt opp denne maskinen selv. Hva vil være det beste alternativet for fjerntilgang når maskinen er Linux-basert? (Det finnes andre alternativer, men velg det beste i listen under).

- Velg ett alternativ:
  - ☐ Remote Desktop (RDP)
  - ☐ telnet
  - ☐ UART-kabel
  - ☐ SSH
  - ☐ SSH som rotbruker
  - ☐ telnet som rotbruker
  - ☐ Remote Desktop (RDP) som
  - ☐ domene-administrator

## Oppgave 38

Hva bør vi minimum forvente fra et sanntidsoperativsystem, som eksempelvis BlackBerry QNX Neutrino og Microsoft Azure RTOS ?

- Velg ett alternativ:
  - ☐ Deterministisk oppførsel.
  - ☐ Lavt effektforbruk.
  - ☐ Støtte for svært mange ulike maskiner og arkitekturer.
  - ☐ Høg gjennomsnittlig ytelse.

## Oppgave 39

Hvilken av disse nettverkskonfigurasjonene har best sanntidsegenskaper i et multimasterscenario:

- Velg ett alternativ:
  - ☐ USB med hub og 3 enheter.
  - ☐ RS485 med 10 enheter på nettverket.
  - ☐ Ethernet med 2 enheter på nettverket.
  - ☐ CAN med 24 enheter på nettverket.

## Oppgave 40

Hva er den beste beskrivelsen av bus-arbitrering:

- Velg ett alternativ:
  - ☐ Rekkefølgen nodene på bus-en starter i (slik at vi kan styre om dette er tilfeldig eller ikke).
  - ☐ Fremgangsmåten for å få tilgang til delt kommunikasjonskanal.
  - ☐ Antall ledere på bus-en, og hvordan hver av disse multiplekseres.
  - ☐ Monteringstype, f.eks. Wago-klemmer.

## Oppgave 41

I konteksten operativsystemer. Hva beskriver en preemptive scheduler best?

- Velg ett alternativ:
  - ☐ En scheduler som sørger for færrest mulige avbrudd.
  - ☐ En scheduler som sørger for høyest mulig gjennomsnittlig throughput.
  - ☐ En scheduler som alltid når alle tidsfrister.
  - ☐ En scheduler som kan avbryte oppgaver som kjører

## Oppgave 42

Hva er et eksempel på sanntidssystem av typen firm?

- Velg ett alternativ:
  - ☐ Sanntids-streaming av akselerasjonskommandoer til industrirobot.
  - ☐ Sanntids-streaming av video hvor det ikke er så farlig at én video-frame droppes.
  - ☐ Styringssystem for balansering av en kort invertert pendel.
  - ☐ Styringssystem for liten og lett drone.
  - ☐ Styringssystem for et åpent ustabilt jagerfly.

## Oppgave 43

En av hovedforskjellene mellom de to bussene I2C og SPI er:

- Velg ett alternativ:
  - ☐ SPI er adressert, I2C er IKKE adressert.
  - ☐ I2C er adressert, SPI har dedikerte linjer for å velge enhet master kommuniserer med.
  - ☐ SPI er multi-master, I2C styrer hvem som er master med et eget signal.
  - ☐ I2C er full duplex, det er IKKE SPI.

## Oppgave 44

Hva karakteriserer en småsignaldiode?

- Velg ett alternativ:
  - ☐ Strøm kan kun gå i én retning av gangen.
  - ☐ Strøm kan kun gå i én retning.
  - ☐ Spenningen øker når strøm går gjennom en småsignaldiode.
  - ☐ Dioden kan bli ødelagt hvis du setter opp en svært lav spenning i feil retning

## Oppgave 45

Hva er formålet med å plassere (raske) kondensatorer på supplyspenningen nær en mikrokontroller?

- Velg ett alternativ:
  - ☐ Muliggjøre høyere langvarig strømtrekk på mikrokontrollerens utganger.
  - ☐ Unngå spenningsregulator i kretsen.
  - ☐ Unngå unødvendig elektromagnetisk stråling og oppnå mest mulig stabil spenning inn på mikrokontrolleren.
  - ☐ Unngå dioder nær mikrokontrolleren

## Oppgave 46

Du skal forsyne en mikrokontroller med 3.3 volt, og har kun et knappecellebatteri med 1.2 V i kretsen. Det er ikke plass til andre typer batterier i dette nye og miniatyriserte måleinstrumentet som konstrueres. Hvilken type spenningsregulator velger du for forsyning av mikrokontrolleren?

- Velg ett alternativ:
  - ☐ Buck-converter.
  - ☐ Lineær spenningsregulator.
  - ☐ Boost-converter.
  - ☐ Lineær LDO (Low Drop Out) spenningsregulator.

## Oppgave 47

Illustrasjonen viser et utsnitt fra databladet til en lysdiode (LED). LED-en skal kobles opp slik at den lyser konstant med maksimal effekt i henhold til databladet (uten bruk av PWM eller lignende). Forsyningsspenningen er 12 V. Hvilken spesifikasjon velger du ved bestilling av den strømbegrensende motstanden?

**Electrical and Optical Characteristics (T<sub>A</sub> = 25°C)**

Chip			Lens	Absolute Maximum Ratings				Electro-Optical-Data's at 20mA					Viewing Angle 2 θ ½ (deg)
Emitted Colour	Peak Wavelength λ <sub>p</sub> (nm)	Dominant Wavelength λ <sub>p</sub> (nm)		Δλ	Pd	If	Peak If	Forward Voltage Vf (V)		Luminous Intensity I <sub>v</sub> (mcd)			
				(nm)	(mW)	(mA)	(mA)	typ.	max	min.	typ.	max.	
Blue	470	470	water clear	-	123	35	110	3.2	3.5	4880	7300	9750	15°

\*Peak Forward Current (1/10 Duty Cycle, 10μs Pulse Width)

**Absolute Maximum Ratings (T<sub>A</sub> = 25°C)**

Reverse Voltage	5V
Reverse Current (V <sub>R</sub> = 5V)	≤50μA
Operating Temperature Range	- 30°C ~ +85°C
Storage Temperature Range	- 40°C ~ +100°C
Lead Soldering Temperature (3mm below the body)	265°C for 10 seconds

- Velg ett alternativ:

- ☐ 1/4 W, 220 ohm
- ☐ 1/4 W, 380 ohm
- ☐ 1/2 W, 120 ohm
- ☐ 1/2 W, 260 ohm
- ☐ 1/2 W, 251.48 ohm

## Oppgave 48


```
uint8_t i = 0U;

while(i < 5U) {
    addItem();
    i++;
}

std::cout << +i << std::endl;
```

Hvor mange runder går while-løkken i figuren? Merk at du ikke får vite hva funksjonen addItem gjør, men det spiller ikke noen rolle her. Programmet kompilerer og kjører uten problemer. Svar: while-løkken kjører  runder før betingelsen blir false.

## Oppgave 49



```
#include <vector>

#include <spotrobot.h>
#include <halodirobot.h>

int main()
{
    SpotRobot dawg;
    HalodiRobot eve;

    std::vector<ISayHello*> robots;

    robots.push_back(&dawg);
    robots.push_back(&eve);

    for (auto robot : robots)
    {
        robot->sayHello();
    }
}
```

Roboten Eve kan foreløpig kun hilse ved å vinke. Samtidig har MIL kjøpt in Boston Dynamicsroboten Spot. Den sistnevnte har dessverre ingen hale å hilse med, men den kan bjeffe. Din oppgave er nå å lage et interface i C++ for å representere hilsing. Da kan både Eve-klassen og Spot-klassen arve dette interfacet, slik at vi kan få begge robotene til å hilse ved hjelp av en for-løkke.. Se et eksempel på et program som BRUKER dette interfacet i illustrasjonen. Hva karakteriserer et interface (en abstrakt klasse) i C++ ?

- Velg ett alternativ:
  - ☐ Inneholder IKKE pure virtual destructor.
  - ☐ Kan kun inneholde én funksjon (f.eks. for å hilse, bevege seg etc.).
  - ☐ Inneholder MINST én virtual funksjon.
  - ☐ Inneholder MINST én pure virtual funksjon.



## Oppgave 50

```
#ifndef SPOTROBOT_H
#define SPOTROBOT_H

#include <isayhello.h>

class SpotRobot : ██████████
{
public:
    SpotRobot() = default;
    ~SpotRobot() = default;

    // ISayHello interface
public:
    virtual void sayHello() override;
};

#endif // SPOTROBOT_H
```

Hva mangler her i klassen SpotRobot for å sikre at vi korrekt arver fra interfacet ISayHello. (Som altså er en abstrakt klasse). Velg alternativet som skulle stått der den sorte firkanten sladder C++-koden.

- Velg ett alternativ:
  - ☐ public ISayHello
  - ☐ inherit ISayHello
  - ☐ SayHello
  - ☐ private ISayHello
  - ☐ ISayHello

## Oppgave 51

```
#ifndef SPOTROBOT_H
#define SPOTROBOT_H

#include <isayhello.h>

class SpotRobot : ██████████
{
public:
    SpotRobot() = default;
    ~SpotRobot() = default;

    // ISayHello interface
public:
    virtual void sayHello() override;
};

#endif // SPOTROBOT_H
```

Hva er formålet med de to første og den siste linjen i header-filen vist i illustrasjonen?

- Velg ett alternativ:

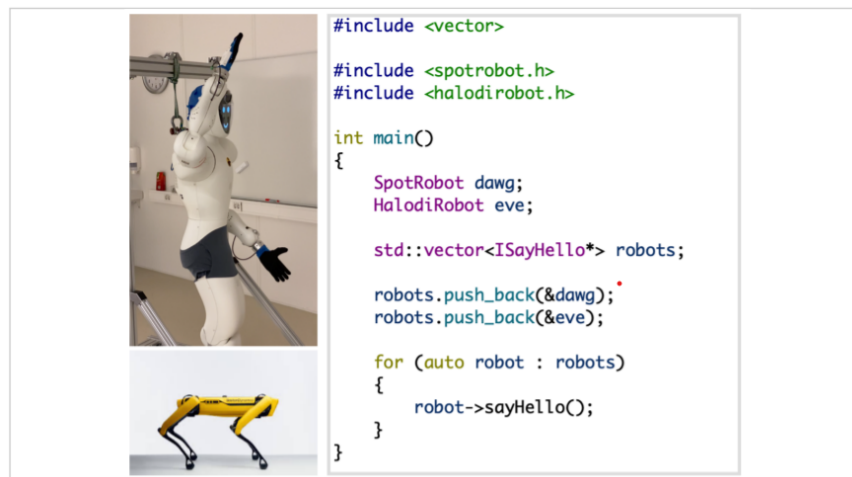
- ☐ Disse linjene er spesifikke for QT Creator, og har ikke noe med C++ å gjøre.
- ☐ Det er en markup som gjør at kompilatoren forstår at dette er en header-fil.
- ☐ Det er en type kommentar som gjør det tydelig hvilken fil dette er.
- ☐ "Include guard", sikrer at en header kun inkluderes én gang per kompileringssammenheng.

## Oppgave 52

Hva beskriver best et API i programmeringssammenheng?

- Velg ett alternativ:
  - ☐ Grensesnittet programmereren må forholde seg til ved bruk av f.eks. et bibliotek eller systemkall i operativsystemet.
  - ☐ Hvordan data plasseres i minnet når data overføres ved funksjonskall.
  - ☐ I Linux er det eneste API-et det som brukes når du kommuniserer med kjernen.
  - ☐ Beskrivelsen av datatyper du kan overføre i et gitt grensesnitt.

## Oppgave 53



Hva gjør de følgende linjene i programmet vist i illustrasjonen? `robots.push_back(&dawg);`  
`robots.push_back(&eve);`

- Velg ett alternativ:
  - ☐ Erstatte elementene på de to første plassene i `std::vector`-en `robots` med henholdsvis `dawg` og `eve`.
  - ☐ Legger inn en referanse til variablene `dawg` og `eve` i `std::vector`-en `robots`.
  - ☐ Legger inn henholdsvis en peker til `dawg` og en peker til `eve` i `std::vector`-en `robots`.
  - ☐ Legger inn en kopi av variablene `dawg` og `eve` i `std::vector`-en `robots`

## Oppgave 54

```
#ifndef SPOTROBOT_H
#define SPOTROBOT_H

#include <isayhello.h>

class SpotRobot : ██████████
{
public:
    SpotRobot() = default;
    ~SpotRobot() = default;

    // ISayHello interface
public:
    virtual void sayHello() override;
};

#endif // SPOTROBOT_H
```

Hva betyr tilde-tegnet først i funksjonsnavnet SpotRobot ?

- Velg ett alternativ:
  - ☐ Invertering av returverdien.
  - ☐ At funksjonen skal arves.
  - ☐ Dette er en destruktør.
  - ☐ Det er vanlig å bruke tilde for å skille to funksjoner med samme navn fra hverandre.