

## **i Informasjon om eksamen**



**Emnekode:** ITF 20415

**Emnenavn:** Systemprogrammering med Windows

**Eksamensdato:** 9. mai 2018

**Tid:** 9.00 til 13.00

**Fagansvarlig:** Einar Krogh

**Tlf.** 69 60 82 99

### **Tillatte hjelpemidler:**

Maks. tre A4-ark med selvskrevne notater.

### **Eksamensoppgaven:**

Oppgavesettet består av 13 oppgaver, samt et vedlegg.

Kontroller at oppgavesettet er komplett før du begynner å besvare spørsmålene.

Oppgavene teller som angitt ved sensurering.

### **Sensurfrist: 30. mai 2018**

Karakterene er tilgjengelige for studenter på Studentweb.

## 1 Oppgave 1 (6 %)

Forklar hvordan Windows operativsystem behandler tråder.  
Hva menes med prosessplanlegging (Scheduling) av tråder?  
Gi eksempel på noen ulike strategier som kan benyttes ved prosessplanlegging.

**Skriv ditt svar her**

## 2 Oppgave 2 (7 %)

Hvordan foregår kommunikasjon mellom maskinvare og brukerprogrammer på en datamaskin.

Hva gjør kommunikasjon mellom brukerprogrammer og maskinvare til et problem?

Hva er oppgaven til Hardwarelaget (HAL) i Windows operativsystem?

**Skriv ditt svar her...**

### 3 Oppgave 3 (6 %)

Gi en oversikt over fenomenet vranglås (Deadlock). Gi gjerne eksempler.

Hvordan kan operativsystemet unngå eller rette opp vranglås?

**Skriv ditt svar her...**

#### 4 Oppgave 4 (6 %)

Hvordan kan man på best mulig måte utnytte at en datamaskin har flere prosessorer.

Nevn noen måter som organisering av bruk av flere prosessorer har vært forsøkt på.

En av dem er Symmetric Multi Processing (SMP). Beskriv hvordan SMP fungerer.

**Skriv ditt svar her...**

## 5 Oppgave 5 (6 %)

I de siste versjonene av Windows Server består Active Directory av fem ulike verktøy.

Gi en beskrivelse av oppgaven til disse verktøyene.

Active Directory Domain Services (AD DS) har hatt en sentral oppgave ved administrasjon av lokalnettverk.

Beskriv hva AD DS benyttes til.

**Skriv ditt svar her...**

## 6 Oppgave 6 (6 %)

Til Raspberry Pi kan det knyttes mye ulikt maskinvare utstyr.

Nevn noen typer utstyr man gjerne knytter til Raspberry Pi.

Beskriv hvilke muligheter Raspberry Pi har til å kommunisere med tilknyttet utstyr.

**Skriv ditt svar her...**

## 7 Oppgave 7 (6 %)

Hvilke operativsystemer kan benyttes på Raspberry Pi?

Gi en nærmere beskrivelse av operativsystemet Windows 10 IoT Core.

**Skriv ditt svar her...**



## 8 Oppgave 8 (6 %)

Hvilke operativsystemer benyttes i innebygde systemer (Embedded Systems).

Hvilke forskjeller er det på operativsystemer som benyttes i innebygde systemer og operativsystemer laget for en PC?

**Skriv ditt svar her...**

## 9 Oppgave 9 (6 %)

Gi en definisjon av hva som menes med nettskyen (Cloud Computing).

Hva er årsaken til at fler og fler bedrifter tar i bruk nettskyen?

Gi en oversikt over fordeler og ulemper med å benytte nettskyen.

**Skriv ditt svar her...**

## 10 Oppgave 10 (7 %)

Gi en beskrivelse av følgende begreper som benyttes i forbindelse med tingenes internett:

Ting, IoT Gateway, sensor, aktuar, Edge Computing, Middleware, Firmware, Big Data, Front End og Back End.

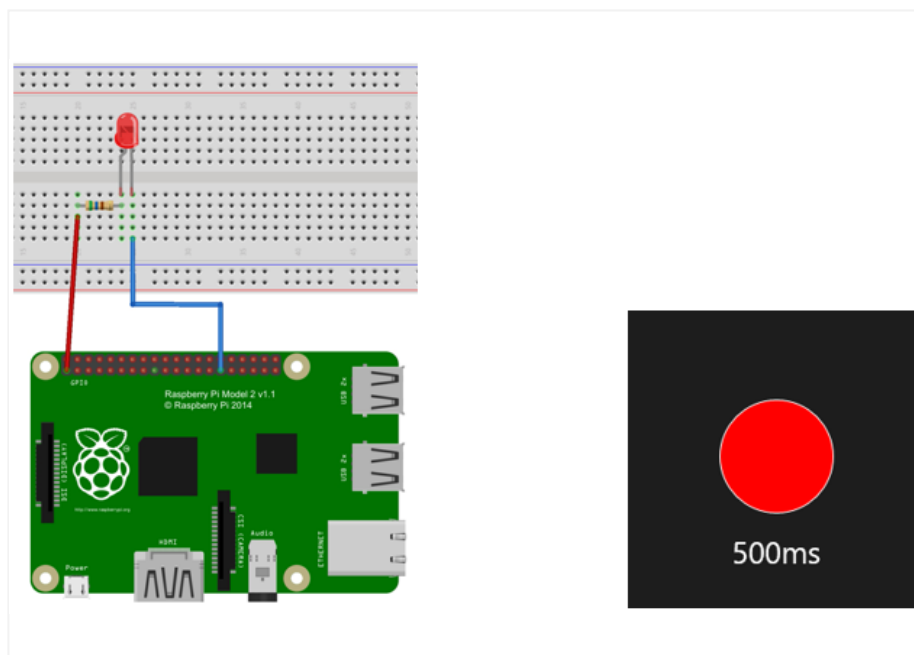
**Skriv ditt svar her...**

## 11 Oppgave 11 (6 %)

Gi en beskrivelse av hvordan tingenes internett (IoT) fungerer. Gi gjerne et praktisk eksempel.

**Skriv ditt svar her...**

## 12 Oppgave 12 (16 %)



Bildet til venstre viser en LED lampe tilknyttet Raspberry Pi. Skriv koden til et C# program i Visual Studio for Windows 10 IoT Core som får LED lampen til å blinke. Bruk en timer til å få lampen til å blinke regelmessig. Du kan få bruk for følgende funksjoner

```
public MainPage() // konstruktør til klassen
```

```
private void InitGPIO()
```

```
private void Timer_Tick(object sender, object e)
```

Bildet til høyre viser hvordan lampen blinker på skjermen. Ta også med kode som simulerer blinkingen på skjermen i programmet.

**Skriv ditt svar her...**

## 13 Oppgave 13 (16 %)

Skriv et C++ program (Console i Visual Studio) med to tråder som kommuniserer med hverandre via et buffer. Den ene tråden (skrivetråden) skal skrive meldinger til bufret. Den andre tråden (lesetråden) skal lese meldingene fra bufret. Som buffer benyttes en String.

```
private: static String^ _Buffer;
```

Når en tråd har skrevet til / lest fra bufret skal den sove noen sekunder. Bruk class Random for å trekke tilfeldige tall. Lesetråden skal ikke lese bufret før noe er skrevet og skrivetråden skal ikke skrive noe nytt før forrige melding er lest. For å synkronisere lesing/skriving benyttes to event objekter

```
private: static EventWaitHandle^ _skrevetEvent;
```

```
private: static EventWaitHandle^ _lestEvent;
```

Lag fornuftige utskrifter til skjermen fra de to trådene slik at vi hele tiden er oppdatert på hva som skjer. Se vedlegg for noen aktuelle bibliotekrutiner i programmet.

**Skriv ditt svar her...**

## i Vedlegg

Vedlegg til oppgave 13.

### Thread Members

Thread(ThreadStart)	Initializes a new instance of the Thread class, specifying a delegate that allows an object to be passed to the thread when the thread is started.
Start()	Causes a thread to be scheduled for execution.
Sleep()	Blocks the current thread for the specified number of milliseconds.

### Random Members

Random()	Initializes a new instance of the Random class.
Next(Int32, Int32)	Returns a random number within a specified range.

### EventWaitHandle Members

EventWaitHandle(Boolean, EventResetMode)	Initializes a new instance of the EventWaitHandle class, specifying whether the wait handle is initially signaled, and whether it resets automatically or manually.
WaitOne()	Blocks the current thread until the current WaitHandle receives a signal.
Set()	Sets the state of the event to signaled, allowing one or more waiting threads to proceed.