

EKSAMENSOPPGAVE (Deleksamen 1)

Emne: IRE12011 / Programmering og mikrokontrollere

Lærer: Åge T Johansen / 482 49 109

Grupper: 13ELE + 13ELEY	Dato: 18.12.2013	Tid: 09.00 – 12.00
Antall oppgavesider: 5	Antall vedleggsider: 8 Vedlegg 1: 20 flervalgsspørsmål (4 sider) Vedlegg 2: MBED oversikt (4 sider)	
Sensurfrist: 21.01.2014		
Hjelpemidler: Lærebøker: "Mike McGrath: C Programming", "Mike McGrath: C++ Programming" PC med blant annet Code::Blocks finnes tilgjengelig på eksamensrommet. Kalkulator		
KANDIDATEN MÅ SELV KONTROLLERE AT OPPGAVESETTET ER FULLSTENDIG		
Alle deloppgaver teller likt ved bedømming, bortsett fra i flervalgsoppgaven, der 4 spørsmål teller som 1 deloppgave.		
PC uten nettilknytning kan benyttes som hjelp for å løse oppgavene. Alle besvarelser skal leveres på papir (utleverte eksamensark med gule og blå kopier).		
Ønsker du å levere programmer du har laget på utlevert minnepenn, kan du gjøre dette i en konvolutt der du påfører eksamensnummeret. Ved tvil/klage er det papirkopien som gjelder.		
I alle programmeringsoppgaver kan det, etter eget ønske, innføres egne funksjoner, selv om det ikke står noe om dette i oppgaveteksten.		

Oppgave 1 PC-programmer

Formelen for å beregne BMI (Body Mass Index).er gitt som:

$$BMI = \frac{vekt_i_kg}{lengde_i_m^2}$$

- a) Lag en BMI-kalkulator, der programmet spør etter høyde og vekt, beregner BMI og skriver resultatet til skjerm med passende informasjonstekst. Bruk *float* som datatype for variablene du vil benytte.

Skriv C-programmet som løser oppgaven.

Oppgave 2 PC-programmer

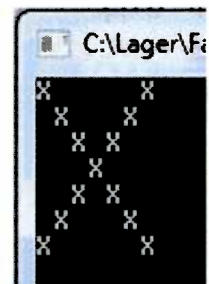
Studer programmet nedenfor. Programmet inneholder både syntaksfeil og funksjonsfeil (semantiske feil).

```
1  #include <stdio.h>
2  #define MAX 7
3
4  int main() {
5      int n, m;
6      for (n = 1; n <= MAX; n++) {
7          for (m = 1, m <= MAX, m++) {
8              if (m == n || m == (MAX+1-n))
9                  {
10                     putchar("X");
11                 }
12                 else
13                 {
14                     printf(" ");
15                 }
16             }
17         }
18     return 0;
19 }
```

Et forsøk på kompilering ga dette resultatet:

File	Line	Message
C:\Lager\Fag\P...		In function 'main':
C:\Lager\Fag\P... 6	6	error: expected expression before '=' token
C:\Lager\Fag\P... 7	7	error: expected expression before ')' token
C:\Lager\Fag\P... 10	10	warning: passing argument 1 of 'putchar' makes integer from pointer without a cast...
c:\program fil... 393	393	note: expected 'int' but argument is of type 'char *'
=== Build finished: 2 errors, 1 warnings (0 minutes, 0 seconds) ===		

Et program der alle feil er rettet opp, skal gi utskriften vist til høyre:



- a) Finn alle syntaksfeil og elementer som vil føre til "warnings" i programmet. Referer til linjenumre, forklar hva feilen / "warning-en" består i og foreslå tiltak for oppretting.

Etter at syntaksfeilene er rettet, vil fortsatt ikke programmet virke etter spesifikasjonen (se figur ovenfor), men gi følgende utskrift som resultat av kjøring:

```

X X X X X X X X X X X X X X X X
Process returned 0 (0x0)   execution time : 0.013 s
Press any key to continue.

```

- b) Vis hvordan programmet kan rettes opp slik at korrekt utskrift oppnås. Skriv programmet på nytt, eller forklar spesifikt med referanse til linjenumre, hva du må gjøre for å rette opp.

Oppgave 3 PC-programmer

I en geometrisk rekke har leddene formen $a_n = a_1 k^{n-1}$, der a_1 er første ledd og k er multiplikasjonsfaktoren.

Summen av de n første leddene finnes av formelen: $S_n = a_1 \frac{k^n - 1}{k - 1}$.

Følgende program beregner summen av 30 ledd der $a_1 = 1.5$ og $k = 1.1$:

```

1  #include <stdio.h>
2
3  int main()
4  {
5      int n = 30;
6      float k = 1.1, a1 = 1.5;
7      float potens = 1.0, sum;
8
9      while (n > 0) { // finner k^n
10         potens = potens * k;
11         n--;
12     }
13     sum = a1*(potens - 1)/(k-1);
14
15     printf("Summen av leddene er: %.2f", sum);
16     return 0;
17 }

```

- a) Skriv nå en C-funksjon som på en generell måte beregner summen av en vilkårlig geometrisk rekke med et vilkårlig antall ledd. Benytt programmet ovenfor som en mal for løsningen, men funksjonen du skriver, må også kunne bregne riktig verdi hvis $k = 1.0$. Funksjonsprototypen til funksjonen skal være som følger:

```
float sum_geo_rekke(int n, float k, float a1);
```

- b) Skriv nå en main()-funksjon som lar brukeren skrive inn antall elementer i rekka, verdien til k og verdien til a_1 fra tastaturet. (Husk ledetekster.) Deretter skal summen beregnes ved å kalle opp funksjonen fra forrige punkt. Til slutt skrives resultatet ut.

Oppgave 4 Flervalgsoppgave

Vedlegg 1 inneholder en flervalgsoppgave med 20 spørsmål. Du skal kun velge **ett** svaralternativ på hvert spørsmål. Velg alternativet du mener **passer best**. Presenter svarene med samme referanser som i vedlegget, med nummer på spørsmålet og bokstav på svaralternativet. Ordne besvarelsen i en tabell som vist nedenfor.

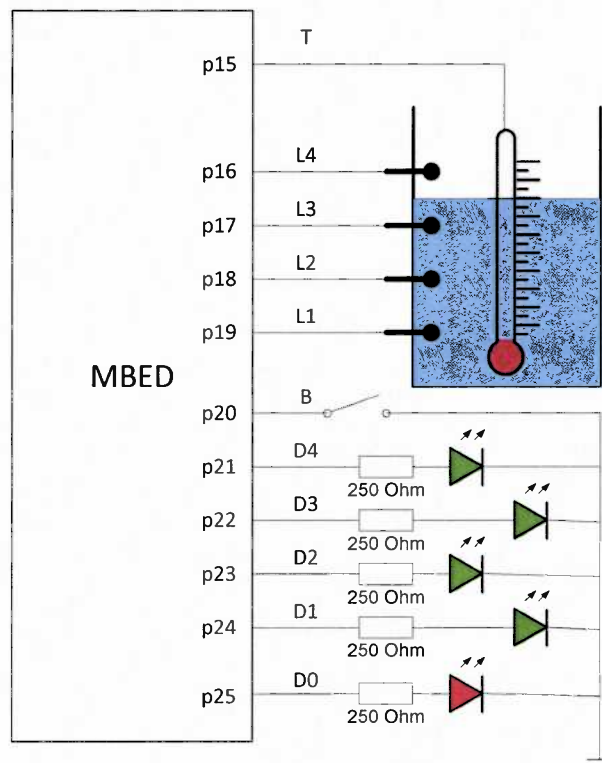
(NB! Svaralternativene som er skrevet inn, er kun eksempler.)

Spørsmål	Svar	Spørsmål	Svar	Spørsmål	Svar	Spørsmål	Svar
1	C	6	...	11	...	16	...
2	A	7	...	12	...	17	...
3	E	8	...	13	...	18	...
4	...	9	...	14	...	19	...
5	...	10	...	15	...	20	...

Oppgave 5 MBED-programmer

Deloppgavene a) – c) refererer til figuren nedenfor som viser et MBED-system. Systemet skal overvåke væskenivået og temperaturen i en tank, som skissert.

- ✓ De digitale følerne L1 – L4 leses av som 1 når de er dekket av væske og som 0 når de er tørre. L1 står for $\frac{1}{4}$ tank, L2 for $\frac{1}{2}$ tank, L3 for $\frac{3}{4}$ tank og L4 for full tank.
- ✓ Temperaturføren T gir en analog spenning mellom 0 V - 3.3 V for temperaturområdet 0°C - 100°C.
- ✓ Bryteren B legges til 0 V når den aktiveres. Intern pullup-motstand må derfor kobles inn når denne skal benyttes for å få avlest en sikker 1 i ikke-aktivert tilstand.
- ✓ Lysdiodene lyser som gitt av den viste koblingen.



Skisse av MBED-system

- a) Studer programmet nedenfor. Hva gjør programmet? (Hva er tilstanden i tanken når dioden lyser – forklar?)

```
#include "mbed.h"

DigitalOut D0(p25);
DigitalIn L1(p19);

int main() {
    while (1) {
        if (L1 == 0) {
            D0 = 1;
        }
        else {
            D0 = 0;
        }
        wait(0.1);
    }
}
```

- b) Skriv et MBED-program som leser av nivåfølerne til tanken og tenner lysdiodene etter følgende spesifikasjon:

- ✓ Når tanken er full skal D4 lyse og ingen andre dioder.
- ✓ Når tanken er $\frac{3}{4}$ full skal D3 lyse og ingen andre dioder.
- ✓ Når tanken er $\frac{1}{2}$ full skal D2 lyse og ingen andre dioder.
- ✓ Når tanken er $\frac{1}{4}$ full skal D1 lyse og ingen andre dioder.

Programmet skal kjøre kontinuerlig, med ca. 4 avlesninger per sekund.

- c) I neste program du skal skrive skal både nivå og temperatur måles.

For nivåmålingen skal diodene tennes som beskrevet i forrige deloppgave.

For temperaturmålingen skal lysdiodene tennes etter følgende spesifikasjon:

- ✓ D4 lyser når temperaturen er over 50°C
- ✓ D3 lyser når temperaturen er over 40°C
- ✓ D3 lyser når temperaturen er over 30°C
- ✓ D3 lyser når temperaturen er over 20°C

Når bryteren B = 1 skal temperaturmåling utføres, når B = 0 skal nivåmåling utføres.

Det skal til enhver tid (dvs. i løpet av $\frac{1}{4}$ sekund) være mulig å veksle mellom disse måleprosedyrene.

D0 skal indikere om det utføres temperatur- eller nivåmåling. D0 lyser under nivåmåling, men er slukket under temperaturmåling.

Programmet skal kjøre kontinuerlig, med ca. 4 avlesninger per sekund.

VEDLEGG 1

	Spørsmål		Svaralternativer
1	For å oversette en C-programfil til maskinkode benyttes hvilket program?	A	Editor
		B	Assembler
		C	Kompilator
		D	Simulator
		E	Prossessor
		F	Ingen av alternativene (A - D)
2	I et C-program må man fortelle kompilatoren hvor data kan plasseres ved å benytte egendefinerte navn. En slik navngitt datalagringsplass kalles:	A	datastruktur
		B	datatype
		C	løkke
		D	kontrollstruktur
		E	variabel
		F	Ingen av alternativene (A - D)
3	Programelementer som blant annet <i>if</i> , <i>while</i> , <i>for</i> og <i>switch</i> kalles med en felles betegnelse:	A	kontrollstrukturer
		B	algoritmer
		C	funksjoner
		D	datatyper
		E	datastrukturer
		F	Ingen av alternativene (A - D)
4	En datatype som kan benyttes for å angi at en variabel skal inneholde et heltall er:	A	char
		B	int
		C	long int
		D	unsigned int
		E	Alle alternativene (A - D)
		F	Ingen av alternativene (A - D)
5	En datatype som kan benyttes for å angi at en variabel skal inneholde et tall med desimaldel er:	A	long
		B	int
		C	double
		D	short
		E	Alle alternativene (A - D)
		F	Ingen av alternativene (A - D)

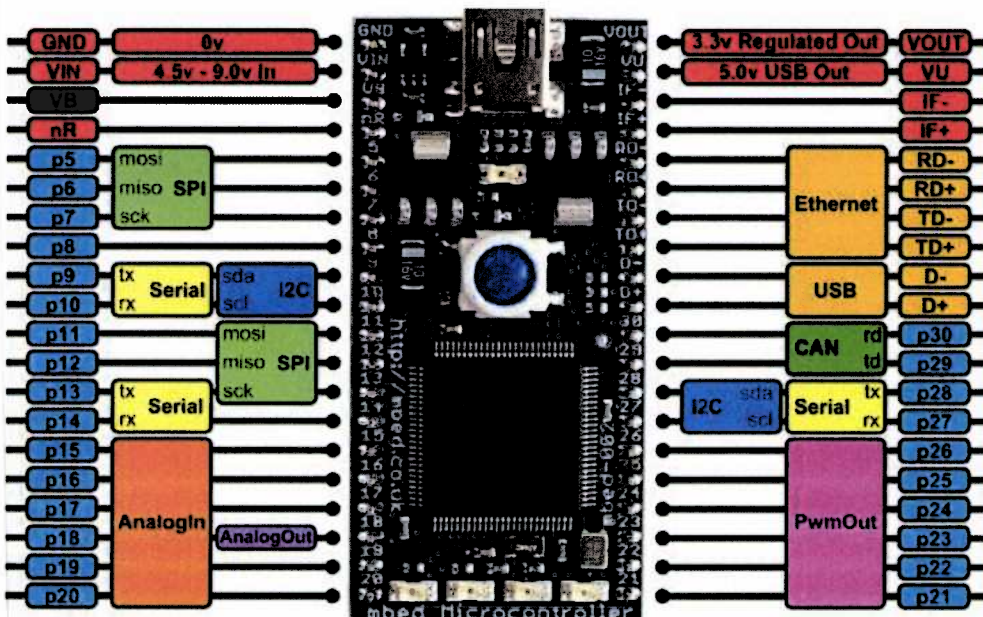
	Spørsmål		Svaralternativer
6	Hva er riktig betegnelse for disse programlinjene: <pre>if (x > valg) { y = 1; } else { y = 5; }</pre>	A	en if-løkke
		B	en if-funksjon
		C	et if-program
		D	en if-test
		E	Alle alternativene (A - D)
		F	Ingen av alternativene (A - D)
7	Hva er riktig betegnelse for denne programlinjen: $x = y + (3 * z) / (z + 1);$	A	en logisk setning
		B	en sammenligningssetning
		C	en aritmetisk setning
		D	en løkke
		E	Alle alternativene (A - D)
		F	Ingen av alternativene (A - D)
8	Hvilke deklarasjon(er) må være gjort på forhånd, hvis følgende 2 løkker skal utføre samme oppgave? <pre>for(i = 5; i < k; i++) { printf("%d ", i); } while(n < 10) { printf("%d ", n++); }</pre>	A	int i, int k, int n;
		B	int i=0, int k=0;
		C	int k=1, int n=5;
		D	int i=0, int k=0, int n=0;
		E	int i, int k=10, int n=5;
		F	int i, int k=10, int j=i;
9	Anta deklarasjonene: <pre>int testtab[10000]; int i; long int l; float f; double d; char c;</pre> Hvilke av de deklarete variablene kan benyttes som indekser i tabellen, hvis alle elementer skal kunne nås? (f. eks. testtab[c])	A	<i>i</i> , <i>l</i> og <i>c</i>
		B	<i>l</i> og <i>c</i>
		C	<i>i</i> , <i>f</i> og <i>c</i>
		D	<i>f</i> , <i>c</i> og <i>i</i>
		E	<i>i</i> og <i>l</i>
		F	<i>d</i> , <i>i</i> og <i>l</i>
10	Hva er resultatet av uttrykket? $(1 != 0) + (10 < 2) + (4 / 2)$	A	0
		B	1
		C	2
		D	3
		E	4
		F	Ingen av alternativene (A - D)

	Spørsmål		Svaralternativer
11	En switch-setning benyttes som oftest hvis man ønsker å kunne:	A	Utføre en aktivitet et bestemt antall ganger.
		B	Utføre én av to alternative aktiviteter
		C	Utføre én av mange alternative aktiviteter
		D	Lese av status for en bryter
		E	Alle alternativene ovenfor
		F	Ingen av alternativene ovenfor
12	Hva er den <i>mest</i> dekkende beskrivelsen av forholdet mellom programmeringsspråkene C og C++?	A	C++ er basert på C
		B	C er basert på C++
		C	C++ og C har ingenting med hverandre å gjøre
		D	C er en forenkling av C++ egnet til begynneropplæring
		E	C++ er en objektorientert utvidelse av C
		F	C++ er en funksjonsorientert utvidelse av C
13	I C++ tilsvarer relasjonen mellom <i>klasser</i> og <i>objekter</i> relasjonen mellom:	A	konstanter og variable
		B	variable og datatyper
		C	konstanter og datastrukturer
		D	datastrukturer og kontrollstrukturer
		E	datatyper og konstanter
		F	datatyper og variable
14	Anta at deklarasjonen <code>unsigned short int x;</code> fører til at en 16-bits variabel opprettes. Hva er den største verdien <i>x</i> kan ha?	A	1023
		B	4095
		C	32767
		D	65535
		E	1048575
		F	4294967295
15	Et standard C-program struktureres ved at det organiseres i programelementer som kalles:	A	funksjoner
		B	ohmegasetninger
		C	blokker
		D	objekter
		E	Alle alternativene ovenfor
		F	Ingen av alternativene ovenfor

	Spørsmål		Svaralternativer
16	Main-funksjonen i et C-program har ofte "return 0;" som siste setning. Hva betyr dette?	A	Gjenta programmet fra start
		B	Avslutt programmet uten utskrift
		C	Avslutt og signaliser til operativsystemet en normal avslutning.
		D	Avslutt og signaliser til operativsystemet en unormal avslutning.
		E	Gjenta programmet fra begynnelsen til beregningene gir svaret 0
		F	Ingen av alternativene (A - E)
17	Se følgende MBED-program der LED1 som vanlig representerer den første blå lysdioden på modulen. <pre>#include "mbed.h" DigitalOut blue1(LED1); int main(){ for (int i=0; i<=10; i++) { blue1 = 1; wait(1.0); blue1 = 1; wait(1.0); blue1 = 0; wait(1.0); i++; } }</pre> Hvor mange ganger blinker dioden?	A	4
		B	6
		C	7
		D	9
		E	10
		F	11
18	Nytt MBED-program. <pre>#include "mbed.h" DigitalOut blue1(LED1); int main(){ for (int n=1; n<5; n++) { blue1 = 1; wait(n); blue1 = 0; wait(n); } }</pre> Hvor mange ganger blinker dioden nå?	A	4
		B	6
		C	7
		D	9
		E	10
		F	11
19	Samme program som i forrige punkt. Hvor lag tid vil det ta å utføre programmet (tilnærmet)?	A	2 s
		B	5 s
		C	10 s
		D	20 s
		E	30 s
		F	50 s
20	Hvis du tar i bruk MBED-klassen Ticker, som vist nedenfor; hva vil det da føre til: <pre>Ticker t; int main() { t.attach(&foo, 10); ... }</pre> (Kun deler av programmet er vist.)	A	Funksjonen <code>foo()</code> vil startes hvert 10. sek.
		B	Funksjonen <code>foo()</code> vil startes én gang om 10 sek.
		C	Funksjonen <code>foo()</code> vil startes 10 ganger
		D	Funksjonen <code>foo()</code> vil startes om 10 millisek.
		E	Funksjonen <code>foo()</code> vil startes 100 ganger
		F	Ingen av alternativene (A - E)

VEDLEGG 2

Oversikt over ofte benyttede ressurser for *mbed*, som beskrevet i "<http://mbed.org/handbook>". (Den engelske teksten er uforandret, men formatet kan være redigert.)



DigitalOut

A digital output, used for setting the state of a pin.

Functions

DigitalOut Create a DigitalOut connected to the specified pin
write Set the output, specified as 0 or 1 (int)
read Return the output setting, represented as 0 or 1 (int)
operator= A shorthand for write
operator int() A shorthand for read

Interface

The DigitalOut Interface can be used on mbed pins p5-p30, and also on-board LED1-LED4 .

The DigitalOut Interface can be used to set the state of the output pin, and also read back the current output state. Set the DigitalOut to zero to turn it off, or 1 to turn it on.

Details

The pin output is 0v and 3.3v (0 and 1), and can source or sink a maximum of 40mA.

DigitalIn

A digital input, used for reading the state of a pin.

Functions

DigitalIn Create a DigitalIn connected to the specified pin
read Read the input, represented as 0 or 1 (int)
mode Set the input pin mode
operator int() An operator shorthand for read()

Details

The pin input is logic '0' for any voltage on the pin below 0.8v, and '1' for any voltage above 2.0v. By default, the DigitalIn is setup with an internal pull-down resistor.

BusOut

The BusOut interface is used to create a number of [DigitalOut](#) pins that can be written as one value.

Example.

```
#include "mbed.h"
BusOut myleds(LED1, LED2, LED3, LED4);

int main() {
    while(1) {
        for(int i=0; i<16; i++) {
            myleds = i;
            wait(0.25);
        }
    }
}
```

BusIn

The BusIn interface is used to create a number of DigitalIn pins that can be read as one value. Any of the numbered mbed pins can be used as a DigitalIn in the BusIn.

Example:

```
#include "mbed.h"
BusIn nibble(p5, p6, p18, p11);

int main() {
    while(1) {
        switch(nibble) {
            case 0x3: printf("Hello!\n"); break; // p5 and p6 are 1
            case 0x8: printf("World!\n"); break; // p11 is 1
        }
    }
}
```

AnalogIn

An analog input, used for reading the voltage on a pin.

Functions

AnalogIn Create an AnalogIn, connected to the specified pin
read Read the input voltage, represented as a float in the range [0.0, 1.0]
read_u16 Read the input voltage, represented as an unsigned short in the range [0x0, 0xFFFF]
operator float An operator shorthand for read()

Details

The AnalogIn Interface can be used on *mbed* pins p15-p20.

The 0.0v to 3.3v range of the AnalogIn is represented in software as a normalized floating point number from 0.0 to 1.0.

AnalogOut

An analog output, used for setting the voltage on a pin

Functions

AnalogOut Create an AnalogOut connected to the specified pin
write Set the output voltage, specified as a percentage (float)
write_u16 Set the output voltage, represented as an unsigned short in the range [0x0, 0xFFFF]
read Return the current output voltage setting, measured as a percentage (float)
operator= An operator shorthand for write()
operator float() An operator shorthand for read()

Details

The AnalogOut Interface can be used on *mbed* pin p18.

The AnalogOut Interface can be used to set the voltage on the analog output pin somewhere in the range of 0.0v to 3.3v.

The 0.0v to 3.3v range of the AnalogOut can be represented in software as a normalized floating point number from 0.0 to 1.0, or directly as volts or millivolts.

wait

Generic wait functions.

Functions:

```
void wait(float s);
```

Waits for a number of seconds, with microsecond resolution (within the accuracy of single precision floating point).

Variables

s - number of seconds to wait

Timer

A general purpose timer

Example:

```
// Count the time to toggle a LED
#include "mbed.h"
Timer timer;
DigitalOut led(LED1);
int begin, end;
int main() {
    timer.start();
    begin = timer.read_us();
    led = !led;
    end = timer.read_us();
    printf("Toggle the led takes %d us", end - begin);
}
```

Functions:

```
void start(); //Start the timer
void stop(); // Stop the timer
void reset(); // Reset the timer to 0. If it was already counting, it will continue
float read(); // Get the time passed in seconds
int read_ms(); // Get the time passed in mili-seconds
int read_us(); // Get the time passed in micro-seconds
```

Ticker

The Ticker interface is used to setup a recurring interrupt to repeatedly call a function at a specified rate. Any number of Ticker objects can be created, allowing multiple outstanding interrupts at the same time. The function can be a static function, or a member function of a particular object.

A Ticker is used to call a function at a recurring interval

Functions

```
attach Attach a function to be called by the Ticker, specifying the interval in seconds
attach Attach a member function to be called by the Ticker, specifying the interval in
seconds
attach_us Attach a function to be called by the Ticker, specifying the interval in micro-seconds
attach_us Attach a member function to be called by the Ticker, specifying the interval in
micro-seconds
detach Detach the function
```