Programming with ANSI C

We are using CodeBlocks C IDE and Compiler. Downloaded from codeblocks.org

The first very simple program.

Most of your programs will have to use two libraries, two standard libraries called `stdio.h` and `stdlib.h`, they contains routines used to input and output data, and standard routines.

Every C program has to contain a function called "`main()`", the compiler will start executing the program from the main() function.

In C we have to declare all used variables with a type and a name.
Examples :

Type	# of BITs	Explanation
boolean	8	simple logical true/false
byte	8	unsigned number 0 - 255
char	8	signed number from -128 to 127.
unsigned char	8	same as 'byte'; Use 'byte' for reasons of clarity
word	16	unsigned number 0 - 6553
unsigned int	16	the same as 'word'. Use 'word' for clarity and brevity
int	16	signed number from -32768 to 32767
unsigned long	32	unsigned number 0 – 4,294,967,295
long	32	signed number from -2,147,483,648 to 2,147,483,647
float	32	signed number from -3.4028235E38 to 3.4028235E3

```
**********************************************************************
#include <stdio.h>
#include <stdlib.h>
int main()
{
  int number;
     char character;
     number = 14;
     character = 'q';
     printf("Printing a character %c and a number %d
\n",character,number);
     return 0;
}
**********************************************************************
```

`printf()`

we are able to format the printout with printf()

Here are some examples:

`printf(”%d”,12);`	12	normal int printout
`printf(”%4d”,12);`	12	4 pos right order
`printf(”%04d”,12);`	0012	4 pos right 0 in front
`printf(”%-4d”,12);`	12	4 pos left order
`printf(”%f”,3.45)`	3.450000	normal float printout
`printf(”%6.2f”,3.4)`	3.40	6 pos with 2 decimals
`printf(”%-6.2f”,3.4569)`	3.46	left with 2 dec.
`printf(”%06.2”,3.4)`	003.40	0 in front
`printf(”%8.2e”,3.456)`	3.46e+00	eksponential
`printf(”%3s”,”abcde”)`	abc	string with 3 pos
`printf(“%6s”,”abcde”)`	abcde	with 6 pos
`printf(“%6.3s”,”abcde”)`	abc	6 pos length 3
`printf(“%-6s”,”abcde”)`	abcde	left 6 pos

Assignment 1.

Try to make a program.

// calculating moms

// you buy 41112 Lego blocks with a price of kr. 11,48 without moms
// make a invoice, where all the numbers are nice lined up under each other.
// Number.
// Price each without moms.
// total cost without moms.
// moms
// total price all inclusive (moms is a tax of 25 %).

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
  //  Place the program here!!!!!!!
   return 0;
}
```

Assignment 2.

Try to make a program where you enter data from the keyboard with the command scanf().

scanf() is hard to use, a little example:
write a string with max 19 characters, followed by a space,
write a character, followed by a space, now a number, followed by a "enter".

```
int number;
char character;
char string[20];
scanf( "%s %c %d", &string, &character, &number);
```

& sign tells that we are using the address of the variable

(**much** more about that later ;-).

Now the we got the values in the program make a printout by help of printf().

Assignment 3.

There are other input metodes, try them.

```
gets(), getchar().
char character;
char name[20];
gets(name);
character = getchar();
```

Assignment 4.

int dage[] ={31,28,31,30,31,30,31,31,30,31,30,31}

Make a program with a ”for loop” to make the printout.

Month nr. 1 contains 31 days

Month nr. 2 contains 28 days and so on.

Assignment 4A.

Make a new program and use a ”while loop” this time.

You can copy the code from assignment 4 and change what you want to change.

Assignment 4B.

Make a new program and use at this time a ”do-while loop”

Assignment 5.

Make a program where you input a character;

If the character is an “a” or “A” the program will printout “Apple”;

If the character is an “b” or “B” the program will printout “Banana”;

If the character is an “c” or “C” the program will printout “Cherry”;

If the character is an “d” or “D” the program will printout “Date”;

If the character is an “e” or “E” the program will printout “Elderberry”;

Otherwise the program will answer “I don’t know any fruit with a ” the entered letter.

Use if-else.

Assignment 5A.

Now try to make the same program with a “switch” command.

Again copy the code from assignment 5 and change to a switch.

Make your own functions.

You are able to write a function with no data transfere into the function, and also no data out of the function, you can work with global variables, variables which can be seen all over in your project.
The function declaration could look like this “void test()”, void here means that there are no return parameters, the empty parenthes after the function name means that there will not be received any input parameters to the function.

Lets make a little example on how to transfere data to and from the function. We want to make a function that will return the smallest of two input numbers.

```
int min(int a, int b)
{
   if a<b
     return a;
   else
     return b;
}


int main()
{
   int x = 15;
   int y = 20;
   printf(“the smallest value is %d\n”, min(x,y));
}
```

Assignment 6.

Make a program where you enter a number into the main function.
Make a function to input a parameter as the number from the main routine.
Inside the function make a math command ex multiply the number by itself, and then return the result.

Let the main function printout the result.

Assignment 6a.

Change the program developed in Assignment 6 so it has a :

main file AND .c an .h file with the procedure

```
int min(int a, int b)
```