

Exercise 1: Creating your CSC215 directory

1. Launch the terminal
 2. It will open in your **home** directory. If not, type the command: `cd ~` and press enter ↵.
 3. To create a new directory with the name "CSC215", type the command: `mkdir CSC215` then ↵.
 4. Type `ls` ↵ to view the current files and folders. You will be able to see your newly created directory.
 5. To enter the directory "CSC215" type: `cd CSC215` ↵.
- Create a new directory with the name "Lab01" inside "CSC215" 1 point

Exercise 2: Writing your first c program

Creating the program file using emacs:

1. While in the terminal, inside the directory "Lab01", type: `emacs hello.c` ↵
This will launch the GNU emacs application, with a new document titles "hello.c"
 2. To save the file on the disk, press the sequence: `CTRL+x CTRL+s`
 3. To close the emacs application, press the sequence: `CTRL+x CTRL+C`
- Reopen the file "hello.c" in emacs 1 point

Writing the program using emacs:

1. Open the file "hello.c" in emacs and type the following c code:

```
#include <stdio.h>
int main(){
    puts("Hello World !\n");
    Return 0;
}
```
2. Save your work.
3. Close the editor.
4. In the terminal, type: `ls` to view your files and make sure that "hello.c" is created and updated.

Exercise 3: Compiling your first c program using GCC

1. While in the terminal, in directory "Lab01", type: `gcc -Wall -ansi -o hello hello.c` ↵
If your program contains no errors this will produce a file: "hello" in the current directory
 2. Run the program hello by typing: `./hello` ↵
- Modify the 4th line in "hello.c" to: `puts("Hello World !");` ; Recompile and run. 1 point
 - Modify the 4th line in "hello.c" to: `printf("Hello World !");` ; Recompile and run. 1 point

Exercise 4: Using printf with char and int arguments

1. Create a new c file named "ex4.c"
2. Type the following program and save it:

```
#include <stdio.h>
int main(){
    char letter = 'b';
    printf("%c\n", letter);
    printf("%d\n", letter);
    printf("%c\t%d\n", letter, letter);
    return 0;
}
```
3. Compile and run. Record your output.
4. Modify the program by adding the following statement right before return line:

```
printf("%c\t%c\n", letter, letter+15);
```
5. Compile and run. Record your output.

- Explain the last result.

1 point

Lab assignment:

5 points

Write a C program that declares a char variable, say, ch, and initializes it to any lowercase letter, ex: ch = 'b'. The program should:

1. print the character ch and
2. print in a new line the three characters that follow the CH character in the alphabetical order.

Note: In your answer don't change the value of ch and don't use any other variable.

Expected output:

```
CH = b
The following three characters are: c d e
```