UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Computer Programming 143

Practical 0

2019

Aim of Practical 0:

Use two assignments (Assignment A and B) to

(a) Introduce students to the available facilities.

(b) Introduction on how to use the Code::Blocks IDE to create C projects.

"In the beginning the Universe was created. This has made a lot of people very angry and been widely regarded as a bad move." - Douglas Adams

"Everything starts somewhere, although many physicists disagree." - Terry Pratchett, Hogfather

Instructions

1. Attendance is **compulsory** for all the practical sessions of your assigned group. See the study guide for more details.
2. The last section (usually the last 30 minutes) of the practical will be used for a test.
3. If more than two tests have been missed for what ever reason, you will receive an **incomplete** for the subject. See the study guide for more details.
4. You must do all assignments **on your own**. Students are encouraged to help each other **understand** the problems and solutions, but each should write his/her own code. By simply copying someone else's code or solutions, you will not build an understanding of the work.
5. You are responsible for your own progress. Ensure that you understand the practical work. Check your work against the memorandum that will be posted on Wednesday afternoons on learn.sun.ac.za.
6. Use H:\CP143 as your Code::Blocks workspace folder for all projects. But it is highly suggested that you also use a **flash drive to backup** all your work.
7. Create a new project **for each assignment**. See *Creating a Code::Blocks Project* on page 9 for instructions on how to do this.
8. Include a comment block at the top of each source file according to the format given. It must include the correct filename and date, your name and student number, the copying declaration, and the title of the source file.
9. **Indent your code correctly.** Making your code readable is not beautification, it is a time and life saving habit. Adhere to the standards (refer to the documents on SUNLearn).
10. Comment your code sufficiently well. It is required for you and others to understand what you have done.
11. A video to help you getting started with the practical will also be available on SUNLearn.

Opening the Code::Blocks IDE

- Once you have logged on to the computer, click on **Start** → **All Programs** → **Development & Programming** → **CodeBlocks** → **CodeBlocks**.
- A **Compilers autodetection** window might pop up. Just select **GNU GCC Compiler** and click **OK**.
- A **File associations** window might pop up. Just select **Yes, associate Code::Blocks with C/C++ file types**, and click **OK**.

Assignment 0A

Goal:

Type, compile and run a C-program, which will display text on the screen.
Find and fix programming errors in Code::Blocks.

1. a) Create a new project with the name **`Assignment0A`** and a source file **`main.c`**. To do this follow the process described in the section headed *Creating a Code::Blocks Project* on page 9.

   b) Change the code created by the template to be similar to the example below. Enter the correct date, your own student number and description.

```c
/* Filename:       main.c
 * Date:           2016/01/01
 * Name:           Doe J.J.
 * Student number: 12345678
 * ------------------------------------------------------
 * By submitting this file electronically, I declare that
 * it is my own original work, and that I have not copied
 * any part of it from another source.
 * ------------------------------------------------------
 * "Hello world!" Program in C
 * ------------------------------------------------------
 */
#include <stdlib.h>
#include <stdio.h>

int main() {
    printf("Hello World!\n"); // Print output
    return 0;             // Terminate program
}
```

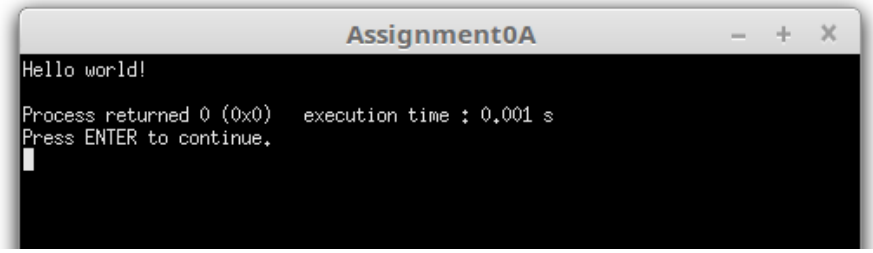
   c) Save the source file by pressing **Ctrl-S**, by selecting **File** → **Save** from the menu bar, or by clicking on the **Save** icon in the toolbar.

   d) Compile the source file by pressing **Ctrl-F9**, or by selecting **Build** → **Build** from the menu bar, or by clicking on the **Build** icon in the toolbar.

   Wait for the compiler to finish.

e) Run the program by pressing **Ctrl-F10**, or by selecting **Build** → **Run** from the menu bar, or by clicking on the **Run** icon in the toolbar.

A new console window should pop up looking similar to this:



2. a) Replace the *printf* statement with the following statement:
   `printf("Hello \nWorld!");`

   b) Compile and run the program. The output should appear over two lines. The **\n** is a command that tells the output stream to insert a carriage return and to start on a new line. Remember this for Assignment B.

3. a) Remove the semicolon at the end of the **printf** statement. Save and compile the source file again.

   b) An error message should appear in the **Build messages** tab at the bottom of your screen. A red block should also appear in the left-hand margin in the window with the source code. This indicates that your code has an **error**, and therefore can't be compiled or executed.

   c) Double-click on the highlighted line in the **Build messages** tag. This should jump to the line in the source code that caused the error. In this case, the compiler only detects the error when it reaches a **return** statement which is not preceded by a semicolon.

   d) Put the semicolon back at the end of the **printf** statement. Save and compile the source file again. The error message and the red block in the code window should disappear.

   e) This feature can be used to find errors in your program: look near red blocks to find errors, or double click an error messages to jump to the error in your code. The fault will often be on the previous line, but not always.

4. Ensure that you copy the **Assignment0A** project folder to a flash drive as a backup.

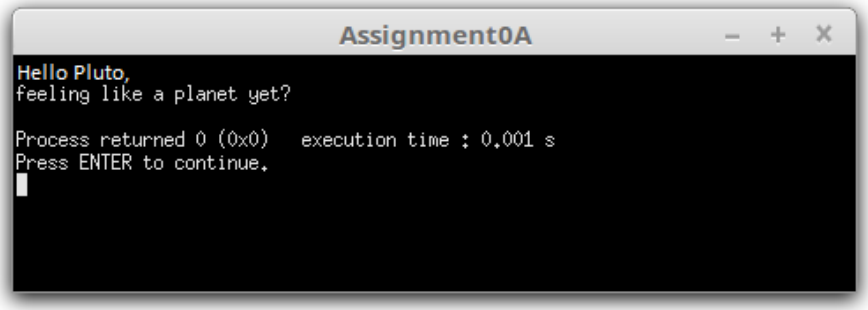
Assignment 0B

Goal:

Type, compile and run a C-program, which will display text on the screen.

1. a) Create a new project with the name **`Assignment0B`** and a source file **`main.c`**. To do this follow the process described in the section headed *Creating a Code::Blocks Project* on page 9. Ensure that the correct project is selected and that the other one is closed before you start programming and compiling.

   b) Write a program that will display the following in the console:



   c) NOTE: Your output must look **exactly** like the above output. That is, the output must be over two lines.

   d) Save the source file by pressing **Ctrl-S**, by selecting **File** → **Save** from the menu bar, or by clicking on the **Save** icon in the toolbar.

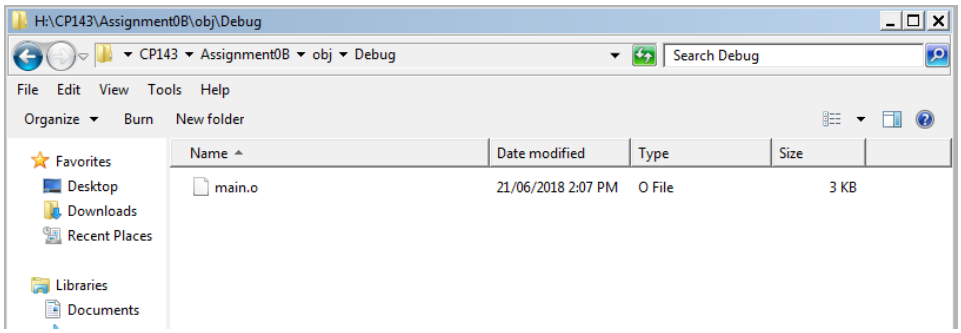
   e) Compile the source file by pressing **Ctrl-F9**, or by selecting **Build** → **Build** from the menu bar, or by clicking on the **Build** icon in the toolbar.

   Wait for the compiler to finish.

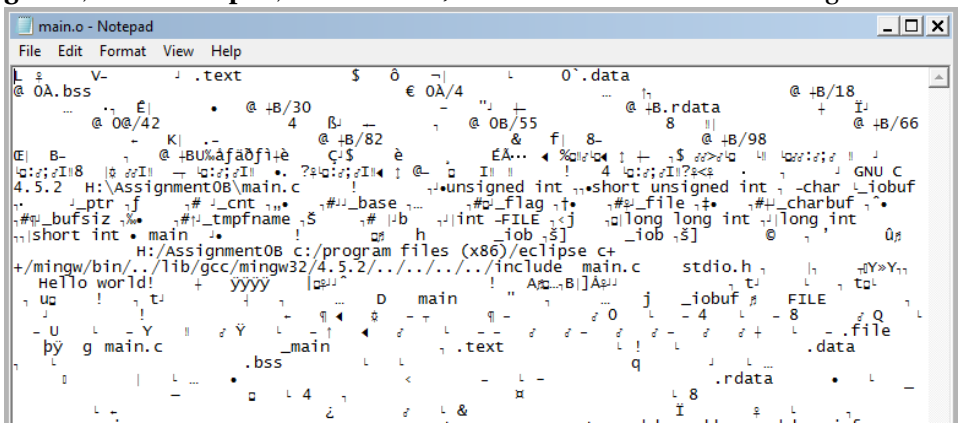
   f) Run the program by pressing **Ctrl-F10**, or by selecting **Build** → **Run** from the menu bar, or by clicking on the **Run** icon in the toolbar.

   The correct message should appear in the console at the bottom of the screen.
   *Tip: You can build and run in one step by just pressing **F9**.*

   g) After successfully running your program, use your file browser (e.g. Windows Explorer) to go to your Code::Blocks workspace folder, go to **Assignment0B** folder, then to the **obj** folder, and finally to the **Debug** folder. The folder content should appear as follows:



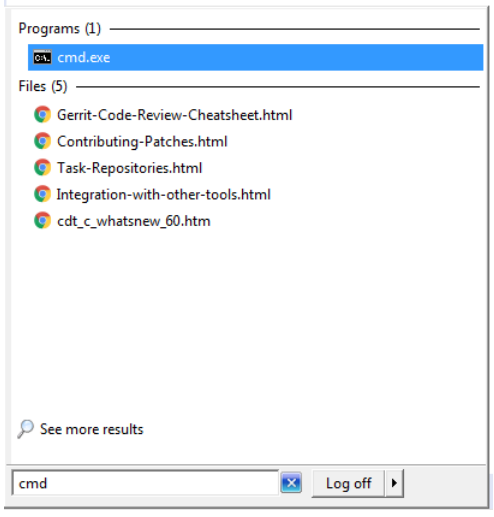
Open the object file (.o) with a text editor like Notepad (e.g. right click on **main.o**, choose **Open**, then select **Select a program from a list of installed programs**, select **Notepad**, and click **OK**). The file should look something like this:



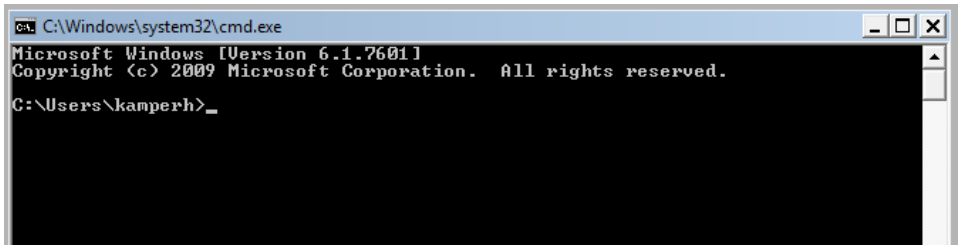
What is the function of an object file, and should it be readable to humans?

h) Now go to the folder **Assignment0B\bin\Debug**. What file do you see there? What is this file?

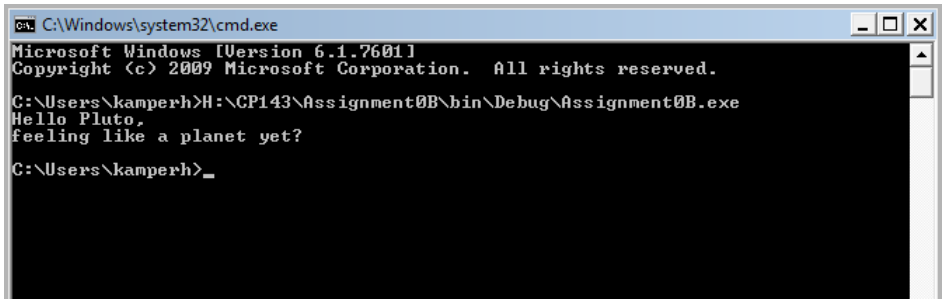
i) Keep the file browser (e.g. Windows Explorer) open. Click on the **Start** button, type `cmd` in the *Search programs and files* text box and press **Enter**.



The Windows terminal screen should appear.



Go to your file browser, click on the executable file **Assignment0B.exe**, and drag it into the Windows terminal screen. The path to this file should appear in the terminal screen. Press Enter and your program should run within the terminal screen.

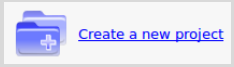


j) Congratulations! You have created your first stand-alone application. This .exe is independent of Code::Blocks and can be used as any other program you have on your computer.

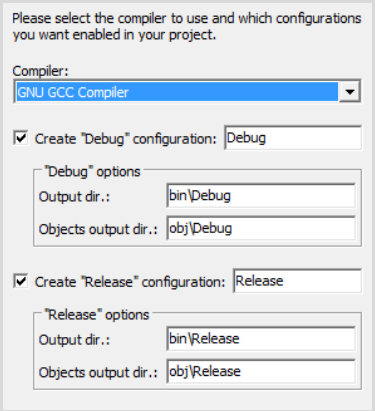
k) Ensure that you copy the **Assignment0B** project folder to a flash drive as a backup.

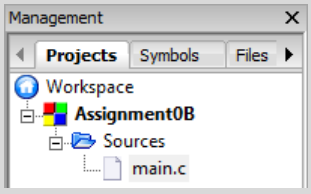
Creating a Code::Blocks Project

Create a new project. *You will do this twice in this practical: once for each question.*

- Click on **File** →**New** →**Project**, or alternatively just click the following button in the **Start here** window: 

- Select **Console application** in the **New from template** window.
- On the window starting with **Welcome to the new console application wizard**, click **Next**.
- Select C as the language you want to use. Click **Next**.
- For the **Project title**, enter **`Assignment0A`** or **`Assignment0B`**, depending on which question you are doing.
- For the **Folder to create project in**, enter **`H:\CP143\`**.
- Leave the rest of the fields as they are. Click **Next**.
- Check that the following screen looks as follows:



- Click **Finish**.
- This will create an empty C project named **`Assignment0A`** or **`Assignment0B`** as the project name, depending on what you entered earlier.
- The new project should be visible in the **Projects** tab under **Management** on the left of the screen: 

- Code::Blocks would have already created a file named **`main.c`** for you. Open this file by clicking on the arrow next to the **Sources** directory in the **Project** tab, and then double clicking on **`main.c`**.
- Some boilerplate code has already been created for you as a starting point. Add a comment block with the date, name of programmer, student number, declaration of own work and a description of what this source file does (refer to the code extract in Assignment0A).

You can now start programming. Have fun!