02393 C++ Programming Exercises

Week 1, February 8, 2016

To be handed in via CodeJudge, before February 8, 5pm
`https://dtu.codejudge.net/02393-f16/assignment/show/1247`

NOTE: The aim of this assignment is to get ready with your C++ installation and to acquire some familiarity with CodeJudge. Do not worry if you are not able to get all exercises right.

In the following exercises, you will have to implement some programs that may need to read some data and then print some result. You can use the following simple example as a template for your program. This simple example just reads a string and prints it.

```
#include <iostream>          // This imports the library iostream,
                             // which contains useful functions for
                             // reading and writing data
using namespace std;         // This allows us to use some abbreviations
                             // For example, we can write cin instead of std::cin

int main()                   // This the main entry of the program
{
    string input;            // We declare a string variable named "input"

    cin >> input;            // We read a string into variable "input"
    cout << input << endl;   // We print the string "input" and a new line (endl).

    return 0;                // This is used to signal correct termination (0)
}
```

- Exercise 1: Write a program that prints "`Hello World!`".
- Exercise 2: Write a program that reads a string (say *name*) from the command line prints "`Hello` *name*`!`". For example, if you run the program and type "`buddy`" the program should print "`Hello buddy!`".
- Exercise 3: Write a program that reads two integer numbers (say, x and y) and prints

  "x `is equal to` y" if x is equal to y;

  "x `is bigger than` y" if x is greater than y;

  "x `is smaller than` y" otherwise.

  For example, if you run the program and type "`1 2`" the program should print "`1 is smaller than 2`".
- Exercise 4: Write a program that reads three numbers (say, x, y and z) and prints number that results from evaluating $(x + y) - z$. For example, if you run the program and type "`1 2 3`" the program should print "`0`".