ECAM
BRUSSELS ENGINEERING SCHOOL

Winter 2020
Sébastien Combéfis

PL190μ Introduction to the Go Programming Language

Coding 4: Iterator design pattern

This assessment evaluates the following competencies:

- *GP001 – Correctly use the syntax of Go programming* *(+1)*
- *GP101 – Write, compile and execute a single source file Go program with the command line* *(+1)*
- *GP002 – Use basic built-in data structure: array, slice and map* *(+2)*
- *GP003 – Manipulate structures and define methods manipulating them* *(+2)*
- *GP301 – Handle rigorously the errors when calling function* *(+2)*

You may also be assessed on the following competencies:

- *GP401 – Understand basic Go compiler errors and warnings and fix the code accordingly* *(+1)*

In this coding assessment, you have to complete an existing Go program that illustrates how to implement the iterator design pattern to ease traversing a slice of `int` [1]. The actual code does compile, but is not correct. First, you have to understand what the code is doing and then, you have to fix it by:

- implementing the `HasNext` and `Next` methods for the `IntSliceIterator` receiver (you can add any necessary field to the `IntSliceIterator` structure;
- implementing the `foo` to test whether your implementation is correct;
- and adapting the code so that the `Next` method produces an error when there is no more elements to inspect, that is, when `HasNext` returns `false`.

To succeed the assessment, you have to explain to the teacher how you implemented the iterator. Then, you have to implement new features, in live, following the specifications given by the teacher, compile the program and run it with the command line. For example, you could be asked to:

- use the iterator on slice of `int` to find whether the slice contains a given value or not;
- add a method to reset the iterator;
- ...

[1] The code can be found here: https://github.com/ukonline/uCourse/blob/master/PL190%C2%B5/code/iterator.go