soft**serve**

Sample of assignments to enter Go Division

Version 1.0

For

***SoftServe IT Academy***

|  |
| --- |
|  |

*January 2017*

Contents

[1 OBJECTIVE 4](#_Toc443489703)

[2 SAMPLES OF THE ENTRANCE TESTS 5](#_Toc443489704)

[3 REFERENCES 6](#_Toc443489705)

**Revision History**

|  |
| --- |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| /01/2017 | 0.1 | Initial version | Mykhaylo Plesha |

# 1 OBJECTIVE

This document has been prepared to make the process of entering SoftServe IT Academy, namely its Go Division easier and more understandable for the candidates. Here there are described the requirements to the level of the candidates’ knowledge, there are given the test samples, a test to determine the level of knowledge and there is given the reference list.

# 2 SAMPLES OF THE ENTRANCE TESTS

**GENERAL**

The entrance test consists of two parts:

1. The first part is theoretical – descriptive set of questions about the OOP
2. The second part is practical — the problem for implementation. To write application code by any known programming language. The validity of the test is checked by teacher assessment manual.

**SAMPLES OF TASKS:**

Sample of assessments for the first part:

1. Basic Concepts of OOPs?
2. What is a class?
3. A set or category of things having some property or attribute in common and differentiated from others by kind, type, or quality.
4. What is an object?
5. What is Encapsulation?
6. What roles perform inheritance?
7. Different between method overriding and method overloading?
8. What are the mechanisms in OOP languages that usually allow provide encapsulation of objects? Why? Show example.

Sample of assessments for the second part:

Write a program that computes and returns the list of Fibonacci numbers (0, 1,  1,  2,  3,..... f(n)=f(n-1)+f(n-2)) which are larger than some number a.

# 3 REFERENCES

* Effective Go: https://golang.org/doc/effective\_go.html
* Caleb Doxsey “An Introduction to Programming in Go”
* Caleb Doxsey “Introducing Go: Build Reliable, Scalable Programs”
* Mark Summerfield “Programming in Go: Creating Applications for the 21st Century”
* Donald Knuth “The Art of Computer Programming”, V.1-4