**STEP by STEP**

***Each construction is more universal and more reliable if it consists of smaller blocks that have been properly tested.***

**The same applies to solving tasks by creating programs for this purpose.**

During our course, we move in this direction step by step and as follows:

1. Each task must be unambiguously understandable - for that, to everyone readable description of the algorithm.

2. Each task must be divided into sub-tasks and certainly in such a way that data input and output are separated from processing.

3. Each task must be tested for extreme situations.

***Using the C language, we proceed as follows:***

1. Let's compile the solution process of the task using, for example, UMli as follows, that the solution process is not related to the C language. At the same time, for each part of the UML in the solution step, we bring a sentence of the C language.

2. We compile a program in C language, as a whole, according to the algorithm.

3. We find and separate possible subtasks in the solution.

4.According to the rules of the C language, we create functions for subtasks.

5. We test all functions separately and the entire program together.

6. We will compileour own library of functions.

consultations on Wednesday U05 104A at 16.00- 18.00 with advance registration vladimir.viies@gmail.com

**Example, first step**

whether it is necessary to find the average value of the measurement results.

1. It must be made clear what type of numbers there are and whether it is known how many there are.
2. UML using „activity“:
3. Here we see three separable activities:

input, processing, output.

1. Program in C language