Python Programming Language

Assoc. Prof. Krzysztof Małecki

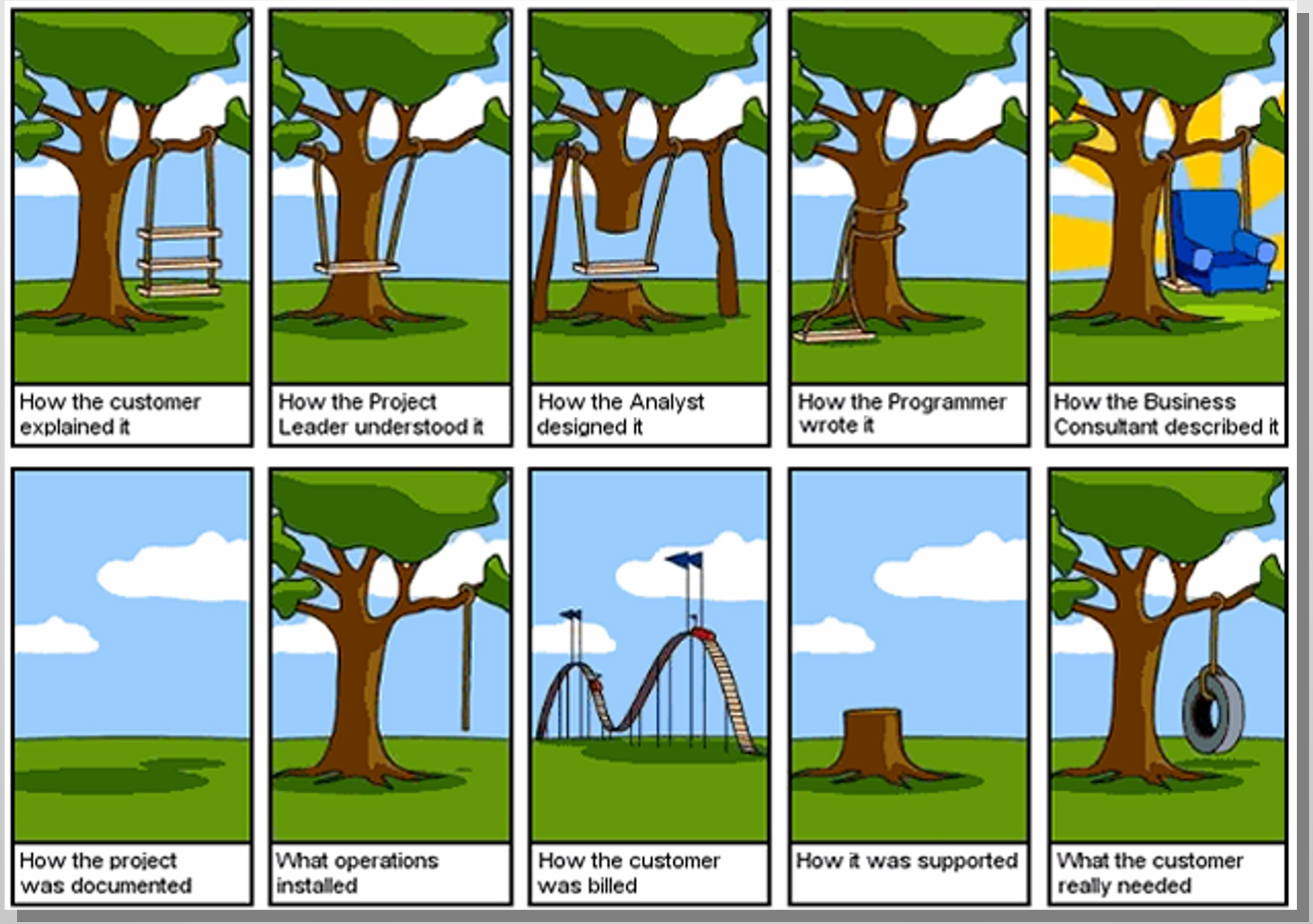
kmalecki@zut.edu.pl
kmalecki.zut.edu.pl
p. 308 (WI2)

Procedure for obtaining credit for the course

- **laboratories**: final assessment based on completed tasks:
  - regular tasks – in every class (the completion of tasks must take place before the next laboratories)
  - the final project – about the last 10-12 hours

- **lectures**: the final test (during the last lecture) – depends on your attendance at the lectures

General meaning of project


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

Let's ask a dramatic question:

What a language is?

Language

is a means of consolidating and expressing thoughts.

So it's time to ask a question:

What is a programming language?

A programming language is a language that serves the formalized representation of algorithms.

What each language consists of?

alphabet → a set of symbols allowed in a given language

(e.g. Latin alphabet in English, Cyrillic in Russian, dots and dashes in telegraph language, etc.)

lexis (dictionary) → a set of such combinations of characters from the alphabet, which in a given language has an assigned meaning

(e.g. the word "spacer" is lexically correct in Polish and English, although it means something else, and the word "ndzjkqwiskj" is not lexically correct in any language)

syntax → a set of such combinations of dictionary elements that are formally correct in a given language

(for example, the expression "*Programmer ate a bread roll*" is syntactically correct, and "*Ate a bread roll programmers*" not)

semantics → a set of syntactically correct expressions that have an assigned meaning (in other words, make sense)

(for example, the phrase "Programmer ate a roll" is semantically correct, and "Roll ate a programmer" not necessarily)

What the program is?

A program is a notation of some algorithm in a specific programming language.

How is the program read?

customarily the program is read and executed:

- from **left** to **right**
- from **up** to **down**

It means:

- what is on the right will follow what is on the left
- what is below will take place after what is above

NOTE - there are **exceptions**!

To be correct, the program must :

- use characters from its alphabet
- meet the requirements of the lexis of the language
- contain syntactically correct constructs
- be in line with the requirements of language semantics
- **do what it was intended for!**

Example:

The programmer was given the task of writing a program that multiplies the numbers **a** and **b**, and stores the result as the number **c**.

The programmer wrote :

```
c = a + b
```

The above program **is** correct because:

- uses the correct alphabet
- it does not contain lexical, syntactic and semantic errors

The above program **is not** correct because :

- does not do what it was assumed for!!!

what is a source code?

- program text stored in a file and intended for compilation or interpretation

- the name of such a file is basically any, but it is assumed that the extension indicates the language that was used to write the program

can we recognize a programming language?

.c	C
.cpp, .cxx	C++
.cs	C#
.pl	Perl
.py, .pyw	Python

- and many many others

what software tools are dedicated to write the source code?

- it must not be written in editors that can format the text - i.e. tools such as MSWord or LibreOffice Writer are categorically excluded!
- it should be written in editors that process the raw text, without any decorations - that is, MS Windows Notepad is OK, but ...

what software tools are dedicated to write the source code?

- there are specialized editors for programmers with a number of useful functionalities, e.g. **Notepad++**, **Sublime Text**
- there are specialized environments (the so-called IDE - Integrated Development Enviroment) which, in addition to the editor, may contain additional tools, e.g. a compiler / interpreter integrated with the editor, i.e. **MS Visual Studio**, **Eclipse**, **PyCharm**, **Spyder**, **Jupyter Notebook** (on-line or off-line)

Thank you for your attention

see you at the next lecture