DESCRIPTION OF THE COURSES

- Course code: ELR1215
- Course title: Visual engineering environments and graphical languages
- Language of the lecturer: English

Course form	***Lecture***	***Classes***	***Laboratory***	***Project***	***Seminar***
*Number of hours/week**	*1*	*0*	*2*	*0*	*0*
*Number of hours/semester**	*15*	*0*	*30*	*0*	*0*
Form of the course completion	*Written test*		*Hands-on test*		
ECTS credits	*1*		*1*		
Total Student's Workload	*30*		*30*		

- Level of the course (basic/advanced): advanced.
- Prerequisites: PC and Windows OS literacy.
- Name, first name and degree of the lecturer/supervisor: Żyłka Paweł, PhD
- Names, first names and degrees of the team's members: --
- Year:....II.... Semester:.....3.....
- Type of the course (obligatory/optional): optional.
- Aims of the course (effects of the course): to introduce students into and familiarise them with the idea of graphical object-oriented programming language; to master their skills in application of graphical engineering environments in control and measurement systems, interfacing and instrumentation.
- Form of the teaching (traditional/e-learning): traditional.
- Course description: complex numerical calculations, designing, servicing and operating various control and measurement systems, interfacing and instrumentation tasks are everyday duties in a contemporary engineer's work. Those engineering actions require practical skills in application of flexible and rapid software development tools. The lecture series is meant to introduce students into the most modern trends in this areas and familiarise them with the idea of graphical object-oriented programming language. The aim is also to to present the advantages, abilities and limitations of the graphical engineering and programming software taking as an example industry-standards: Agillent Vee and NI LabView packages. The following topics are presented: language syntax, data types and structures, data flow rules, basic objects as well as the principles, tips and tricks helpful in designing correct and efficient programs. Emphasise is put on demonstrating and explaining practical applications in measurement, control and test systems as well as in remote data acquisition.
- Lecture:

Particular lectures contents	*Number of hours*
1. Introduction and requirements. Review on the "visual" software packages: high-level languages, process visualisation tools, integrated programming and development environments, graphical languages. Concept of object-oriented graphical programming.	1

2. Data flow, sequencing, multithreading. Variables, registers, containers.	1
3. Data types and structures. Data conversion rules. Functions and user objects. User objects nesting and dependency.	1
4. Basic objects/functional blocks: classification, inputs/outputs, application.	2
5. Advanced mathematical, statistical and signal processing functions.	1
6. Working with text and binary files.	1
7. Data exchange and control of external programs: ActiveX automation, Matlab engine, web server, network solutions.	1
8. Communication and control of external devices. Communication interfaces: RS232/485, GPIB, VXI, Ethernet. I/O transactions.	1
9. Exemplary communication and device control standards: SCPI, VISA, ModBus. IVI device standard.	1
10. Graphical user interface. Virtual instrument idea, *instrument drivers*.	1
11. Principles, tips and tricks helpful in designing data processing and control programs. Efficiency and speed optimisation.	1
12. Special solutions for improving execution speed and numerical processing power. Applications in image processing, fuzzy data sets, adaptation control*, embeded* and *real-time* applications.	1
13. Computer-controlled measurement and test devices and systems – overview.	1
14. Final test.	1

- Classes – the contents: ---
- Seminars – the contents: ---
- Laboratory – the contents: laboratory exercises are meant as a practical illustration of the problems and issues introduced during the lecture. Students, by solving a set of increasingly complex exercises, master their skills and expertise in one of the following graphical engineering environments: Agillent Vee or National Instrument LabView. Classes are run in a computer room, students are also introduced to problems of interfacing of measuring and I/O devices to a PC using parallel and USB ports, to ActiveX automation and database access.
- Project – the contents: ---
- Basic literature:

1. R. Helsel, *Graphical programming - a tutorial for HP Vee*, Prentice Hall PTR, London, 1995.
2. R. H. Bishop, *LabView Student edition 6i*, Upper Sadle River, Prentice-Hall 2001.
3. W. Tłaczała, *Środowisko LabView w eksperymencie wspomaganym komputerowo*, WNT, Warszawa 2002.

- Additional literature:

1. L. U. Wells, *LabView for everyone: graphical programming made even easier*, Upper Saddle River, Prentice Hall 1997.
2. Information available at *Agilent* and *National Instruments* websites, Agillen Vee and NI LabView help files.
3. W. Winiecki, *Organizacja komputerowych systemów pomiarowych*. WPW, Warszawa 1997.

- Conditions of the course acceptance/credition: lecture written final test - positive mark and laboratory hands-on final test – positive mark.

***** - depending on a system of studies