

**ІНСТИТУТ КІБЕРНЕТИКИ
ІМЕНІ В.М. ГЛУШКОВА НАН УКРАЇНИ**



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**ВЕРИФІКАЦІЯ ТА ТЕСТУВАННЯ
(ДВА 2.01.01)**

для здобувачів освітньо-наукового рівня «доктор філософії»

галузь знань	12 «Інформаційні технології»
спеціальність	123 «Комп'ютерна інженерія»
освітній рівень	третій (освітньо-науковий)
освітньо-наукова програма	«Комп'ютерна інженерія»
вид дисципліни	вибіркова

Форма навчання	денна/заочна
Навчальний рік	2020/2021
Рік навчання	2
Кількість кредитів ECTS	4
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	екзамен

Викладач: д.ф.-м.н., ст. наук. співроб. **Летичевський Олександр Олександрович**

Пролонговано Вченою радою Інституту кібернетики імені В.М. Глушкова НАН України

Навчальні роки пролонгації	Голова вченої ради	Підпис	№ протоколу	Дата протоколу
20___/20___ р.	_____	_____	_____	_____
20___/20___ р.	_____	_____	_____	_____
20___/20___ р.	_____	_____	_____	_____
20___/20___ р.	_____	_____	_____	_____

КИЇВ – 2020

РОЗРОБНИК:

Завідувач відділу теорії цифрових автоматів
д.ф.-м.н., старший науковий співробітник

 Летичевський Олександр Олександрович

**Робочу програму розглянуто та схвалено на засіданні відділу
перетворювачів форми інформації**

Протокол від “ 07 ” 07, 20 20 року № 7

Завідувач відділу
д.т.н., професор  В.О. Романов

Робочу програму ухвалено науково-методичною радою

Протокол від “ 15 ” 07, 20 20 року № 3

Голова науково-методичної ради
академік НАН України  І.В. Сергієнко
(підпис)


**Робочу програму затверджено Вченою радою Інституту кібернетики
імені В.М. Глушкова НАН України**

Протокол від “ 28 ” 07, 20 20 року № 13

Учений секретар  А.І. Куляс
(підпис)

**Робочу програму погоджено з гарантом освітньої програми 123 «Комп’ютерна
інженерія»**

“ 12 ” 07, 20 20 року

Гарант освітньої програми
член-кор. НАН України, д.т.н., професор  В.П. Боюн
(підпис)

1.

100 %

2.

1.

2.

3.

4.

5.

(1. ; 2.) ; 3. () ; 4.		(/)	()	
1.1				
1.2				30%
1.3				
2.1				10%

2.2				20%
2.3				20%
3.1				5%
4.1				5%
4.2				10%

6.

	1.1	1.2	1.3	2.1	2.2	2.3	3.1	4.1	4.2
()									
-1.		+	+		+	+			
-7.				+			+	+	+

7.

7.1.

- :
- 1. : PH 1.1, PH 1.2, PH 1.3, PH 2.1 – 10 ;
- 2. : PH 2.2, PH 2.3 – 30 ;
- 3. : PH 2.2, PH 2.3, PH 4.2 – 20 ;
- : ;
- : 40 ;
- : PH 1.2, PH 1.3, PH 2.1, PH 2.3;
- :

7.2. 1 : 1-2, 2 – 3-6, 3 – 7-9.
 , , .
 1. , : ;
 2. : ;
 3. , : .

7.3.

/ Excellent	90-100
/ Good	75-89
/ Satisfactory	60-74
/ Fail	0-59

5	5.	2		12
6	6.	2		12
3.				
7	7.	2		12
8	8.	2		12

	SysML, UCM.			
9	9.	2		12
			4	
		18	4	96

120
-18
-4
-2
-96

9.

1. . -2008
2. : , . . . -2020
3. , .1 -2, . , . , .2000,
4. Paulk Mark C. / (February 1993) / Capability Maturity Model for Software (Version 1.1) / Weber Charles V., Curtis Bill, Chrissis Mary Beth // Technical Report (Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University)
5. ISO/IEC 13568:2002". Information Technology — Z Formal Specification Notation — Syntax, Type System and Semantics // ISO. – 2002. – 196 pp.

6. Bjørner Dines / The Vienna Development Method: The Meta-Language / Cliff B. Jones // Lecture Notes in Computer Science 61. Berlin, Heidelberg, New York: Springer. – 1978.
7. [Gerard Holzmann](#) / The SPIN Model Checker: Primer and Reference Manual // Addison-Wesley – 2004. – 585 pp.
8. [Krzysztof Havel](#) / [UML 2.0](#) (Applying UML 2.0 and Patterns) An Introduction to Object-Oriented Analysis and Design and Iterative Development // — 3- . — .: , 2006. — 736 .
9. Tim Weilkiens / Systems Engineering with SysML/UML: Modeling, Analysis, Design // The OMG Press. –2008.
10. ITU-T Recommendation, Z.151, User Requirements Notation (URN) – Language definition
11. ITU-T Recommendation, Z.120, Message Sequence Charts (MSC)
12. SPIN Homepage, <http://spinroot.com/spin/whatispin.html>
13. Rik Eshuis / Symbolic model checking of UML activity diagrams // ACM Trans. Softw. Eng. Methodol, 15(1). – January 2006. – pp. 1-38.
14. Islam Abdelhalim / Formal verification of tokeneer behaviours modelled in fUML using CSP/ James Sharp, Steve Schneider, and Helen Treharne // In JinSong Dong and Huibiao Zhu, editors, Formal Methods and Software Engineering, volume 6447 of LNCS, Springer Berlin Heidelberg. – 2010. – pp. 371-387.
15. Helle Hvid Hansen / Automated verification of executable UML models / Jeroen Ketema, Bas Luttik, MohammadReza Mousavi, Jaco van de Pol, and Osmar Marchi dos Santos / Proceedings of the 9th International Conference on Formal Methods for Components and Objects, FMCO'10, Berlin, Heidelberg, Springer-Verlag. – 2011. – pp. 225-250.
16. Stefan Blom / Distributed and Symbolic Reachability / Jaco Van De Pol, Michael Weber // Proceedings Computer Aided Verification (CAV2). –2010. –pp. 354-359.
17. Yael Meller / Verifying Behavior UML Systems via CEGAR / Orna Grunberg, Karen Yorav // Proceedings Integrated Formal Methods (IFM). – 2014. – pp.139-154.
18. J. Peleska / Industrial-Strength Model-Based Testing - State of the Art and Current Challenges // Proceedings Eighth Workshop on Model-Based Testing, DOI: 10.4204/EPTCS.111.1. – 2013. – pp. 3-28.
19. Patrice Godefroid / SAGE: Whitebox Fuzzing for Security Testing / Michael Y. Levin, David Molnar // Magazine Queue – Networks, Vol. 10, Issue 1, Jan. – 2012.
20. Willem Visser / Model Checking Programs with Java Pathfinder / Peter Mehlitz // Lecture Notes in Computer Science, Volume 3639. – 2005. – p. 27.
21. Guodong Li / KLOVER: A Symbolic Execution and Automatic Test Generation Tool for C++ Programs / Indradeep Ghosh, Sreeranga P. Rajan // Lecture Notes in Computer Science, Volume 6806. – 2011. – pp. 609-615.
22. S. Ganov / Test Generation for Graphical User Interfaces Based on Symbolic Execution / C. Killmar, S. Khurzid, Dewayne E Perry // Proceedings ISCE Workshop. – 2008.
23. Alexandre Petrenko / Integration Testing of Communication Systems with Unknown Components / Ronald Groz, Keqin Li // Annuals of Communications, vol. 70 , 3-4. – pp 107-125.

24. Gilbert D. / A general theory of action languages / Letichevsky A. // Cybernetics and System Analysis. – 1998. – 1. – pp. 16–37.
25. Gilbert D. / A Model for Interaction of Agents and Environments / Letichevsky A. // Lecture Notes in Computer Science. – 1999. – 1827. – pp. 311–328.
26. Letychevskiy O. / Predicate transformers and system verification / Letichevsky A. // Proc. Third International Workshop on Symbolic Computation in Software Science (SCSS 2010), Hagenberg, Austria. – 2010. A. Voronkov, L. Kovacs and N. Bjorner (eds). EPiC Series, Vol.1. – 2012. – P. 148–149.
27. Letychevskiy O. / Insertion Modeling System / Letichevsky A., Peschanenko V. // Lecture Notes in Computer Science : [Revised selected papers / S.D.J. Brabosa etc]. – Heidelberg : Springer, 2011. – 7162. – P. 262–274