



МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Третій (PhD)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>2 курс, весняний семестр</i>
Обсяг дисципліни	<i>Лекції: 18 год., комп'ютерний практикум: 18 год., самостійна робота: 114 год.</i>
Семестровий контроль/ контрольні заходи	<i>Залік, модульна контрольна робота, блиц-опитування, календарний контроль</i>
Розклад занять	<i>Згідно розкладу на весняний семестр поточного навчального року (rozklad.kpi.ua)</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: к.т.н., доцент, Баклан Ігор Всеволодович, iaa@ukr.net Комп'ютерний практикум: к.т.н., доцент, Баклан Ігор Всеволодович, iaa@ukr.net</i>
Розміщення курсу	<i>http://baklaniv.at.ua/index/multiagentno_orientovane_programuvannja/0-38</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Вивчення дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» дозволяє сформувати у здобувачів освіти компетенції, необхідні для структурований підхід, заснований на трьох взаємопов'язаних наборах концепцій та абстракцій програмування, корисних для проектування таких складних систем: вимір агента, вимір навколишнього середовища та організаційний вимір, що використовується для структурування та регулювання складних взаємозв'язків, що відбуваються між автономними агентами у спільному середовищі.

Метою вивчення дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» є формування у здобувачів освіти здатності провадити науково-інноваційну діяльність, пов'язану із розробленням програмних систем із застосуванням мультиагентного орієнтованого програмування.

Предметом дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» є програмування багатоагентних систем із використанням інтегрованого підходу, який зараз називають мультиагентним орієнтованим програмуванням.

Після засвоєння дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» **результатами навчання** є:

знання:

- отримати змістовний огляд програмування відкритих та децентралізованих систем з використанням мультиагентно орієнтованих перспектив;
- зрозуміння проблем, завдань, що розглядаються в мультиагентному домені

уміння:

- практикувати існуючі мультиагентні технології
- набуття певних навичок мультиагентно орієнтованого програмування;

досвід:

- надання основ, на яких моделях координації можуть бути розроблені MAOC.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Успішному вивченню дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» передую вивчення дисципліни «Формальні методи програмної інженерії» навчального плану підготовки докторів філософії за спеціальністю 121 Інженерія програмного забезпечення.

Отримані в результаті засвоєння дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» теоретичні знання та практичні уміння можуть бути корисні для проведення наукових досліджень за темою дисертації.

3. Зміст навчальної дисципліни

Дисципліна «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» передбачає вивчення таких тем:

Тема 1. Вступ до багатоагентних систем та багатоагентного орієнтованого програмування

Тема 2. Програмування агентів, середовища та організації

Тема 3. Підхід MAOP до координації

Модульна контрольна робота

Залік

4. Навчальні матеріали та ресурси

Базова література:

1. Баклан І.В. Конспект лекцій з дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» для аспірантів за спеціальністю 121 Інженерія програмного забезпечення. – КПІ, 2021.

Використати для опанування теоретичним матеріалом з дисципліни. Матеріали знаходяться на сторінці сайту http://baklaniv.at.ua/index/multiagentno_orientovane_programuvannja/0-38, доступному для зареєстрованих аспірантів.

2. Баклан І.В. Навчальний посібник для виконання комп'ютерного практикуму з дисципліни «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» для аспірантів за спеціальністю 121 Інженерія програмного забезпечення. – КПІ, 2024.

Використати для опанування практичних навичок з дисципліни. Матеріали знаходяться на сторінці сайту http://baklaniv.at.ua/index/multiagentno_orientovane_programuvannja/0-38, доступному для зареєстрованих аспірантів.

Додаткова література:

3. Multi-Agent Oriented Programming: Programming Multi-Agent Systems Using JaCaMo / Olivier Boissier, Rafael H. Bordini, Jomi F. Hübner, Alessandro Ricci. - Massachusetts Institute of Technology — 2020. - ISBN 9780262044578. - 240 pp.

Використати для вивчення мультиагентного орієнтованого програмування на JaCaMo.

4. Balke, Tina, Célia da Costa Pereira, Frank Dignum, Emiliano Lorini, Antonino Rotolo, Wamberto Vasconcelos, and Serena Villata. 2013. Norms in MAS: Definitions and related concepts. In *Normative multi-agent systems*, eds. Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert W. N. van der Torre. Vol. 4 of *Dagstuhl follow-ups*, 1–31. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. <https://doi.org/10.4230/DFU.Vol4.12111.1>.
5. Boissier, Olivier, Rafael H. Bordini, Jomi F. Hübner, and Alessandro Ricci. 2019. Dimensions in programming multi-agent systems. *The Knowledge Engineering Review* 34. <https://doi.org/10.1017/S026988891800005X>.
6. Bosello, Michael, and Alessandro Ricci. 2019. From programming agents to educating agents: A Jason-based framework for integrating learning in the development of cognitive agents. In *Engineering Multi-Agent Systems - 7th International Workshop, EMAS 2019, Montreal, May 11–12*.
7. Cardoso, Rafael C., and Rafael H. Bordini. 2019. Decentralised planning for multi-agent programming platforms. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, Canada, May 13–17, 2019*, eds. Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor, 799–818. <http://dl.acm.org/citation.cfm?id=3331771>.
8. Panisson, Alison R., and Rafael H. Bordini. 2017a. Argumentation schemes in multi-agent systems: A social perspective. In *Engineering Multi-Agent Systems - 5th International Workshop, EMAS 2017, São Paulo, Brazil, May 8–9, 2017, revised selected papers*, eds. Kate Larson, Michael Winikoff, Sanmay Das, and Edmund H. Durfee, 92–108. ACM. https://doi.org/10.1007/978-3-319-91899-0_6.
9. Silva, Lavindra de. 2018b. HTN acting: A formalism and an algorithm. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10–15, 2018*, eds. Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar, 363–371. International Foundation for Autonomous Agents and Multiagent Systems. <http://dl.acm.org/citation.cfm?id=3237441>.
10. Trentin, Iago Felipe, Olivier Boissier, and Fano Ramparany. 2019. Insights about user-centric contextual online adaptation of coordinated multi-agent systems in smart homes. In *Actes des 17èmes Rencontres des Jeunes Chercheurs en Intelligence Artificielle, RJCIA 2019, Toulouse, France, July 2–4, 2019.*, ed. Maxime Lefrançois, 35–42. <https://hal.archives-ouvertes.fr/hal-02160421>.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

№ з/п	Тип навчального заняття	Опис навчального заняття
1	Лекція 1. Вступ до багатоагентних систем та багатоагентного орієнтованого програмування	У цій лекції ми представляємо основні поняття мультиагентних систем (MAS) та надаємо огляд мультиагентного орієнтованого програмування як комплексного підходу до програмування MAS.
2	Лекція 2. Агенти програмування	У цій лекції ми заглиблюємось у деталі першого виміру MAOP, який ми розглядаємо в цій дисципліні, вимір агента. Для початку ми згадуємо загальну картину цього виміру, наведену в попередній лекції. Передається важливість цього виміру в контексті мультиагентного орієнтованого програмування. Потім ми розглядаємо кожну окрему концепцію програмування та абстракцію, пов'язану з цим

		виміром, у структурі JaCaMo, зазначаючи, що більшість із цих концепцій використовується більш загально у MAOP. Після цього ми обговорюємо модель виконання агента, пояснюючи подальші концепції, пов'язані з агентом. Закінчується лекція подальшими примітками, включаючи деякі історичні примітки для слухача дисципліни, зацікавленого у відстеженні подій, що ведуть до сучасного стану технологій програмування.
3	Комп'ютерний практикум 1-7	У комп'ютерному практикумі представляється JaCaMo, конкретну платформу, прийняту в цій дисципліні для практичного мультиагентного програмування. Ця платформа підтримує практичне програмування на основі абстракцій, представлених у лекціях: програмування організованих агентів, розташованих у спільному середовищі. JaCaMo побудований на основі трьох існуючих платформ, які розроблялись роками, а саме Jason для програмістів, CArtaGo для середовищ програмування та Moise для програмних організацій.
4	Лекція 3. Середовище програмування	Після введення виміру агента в попередній лекції поступово збагачуємо наш набір абстракцій програмування, враховуючи вимір середовища, який використовується в MAOP для моделювання ресурсів та інструментів, що використовуються агентами як першокласні концепції. По-перше, представляємо концепцію артефакту, яка є основною цеглою, яку можна використовувати для проектування модульних, динамічних та компонованих середовищ, таких як артефакти в середовищі людини, обговорюючи відповідні абстракції програмування, доступні в JaCaMo. Потім бачимо, як можна програмувати агентів для створення, використання та спостереження артефактів, здатних врешті-решт сформувати своє власне середовище відповідно до своїх потреб та цілей. Нарешті, описується, як складні середовища, засновані на артефактах, можуть бути структуровані з робочими просторами - концепція, введена для моделювання топології багатоагентної системи, можливо розподіленої по декількох вузлах Інтернету.

5	Лекція 4. Програмування агента та його середовища	У цій лекції розглядається на практиці програмування агентів у середовищі, розглядаючи ігровий (але реалістичний) сценарій, який називають розумною кімнатою. У цій лекції ми крок за кроком програмуємо просту одноагентну систему управління температурою кімнати. Тематичне дослідження додатково розширюється кількома агентами, розташованими в одному робочому середовищі, зосереджуючись на програмуванні взаємодії між агентами та на програмуванні агентських організацій. Досліджується, як ми можемо координувати роботу агентів так що вони виробляють узгоджену та спільну глобальну поведінку.
6	Лекція 5. Програмування декількох агентів, що взаємодіють в середовищі	Взаємодія агентів є важливими аспектами багатоагентних систем. Основні механізми, введені в попередніх лекціях про вимір агента та про вимір навколишнього середовища, дозволяють визначити різні складні типи взаємодії. У цій лекції заглиблюємось у ці аспекти, що досліджуються на обох агентах прямої взаємодії, яка базується на мовленнєвій дії, та непряма взаємодія агента, опосередкована через артефакти навколишнього середовища. Розглядаючи їх з практичної точки зору в цьому голові, ми розглядаємо розширення сценарію розумної кімнати з попередньої лекції, яка тепер включає кілька агентів.
7	Лекція 6. Організаційний вимір	У попередніх розділах ми бачили, як розміри агента та середовища забезпечують основні елементи для побудови мультиагентних систем як набору окремих автономних агентів, що спілкуються та працюють у спільних середовищах. У цій лекції ми завершуємо картину MAOP, розглядаючи організаційний вимір, який надає концепції та першокласні абстракції програмування для визначення та управління складними MAS з макро точки зору, порівняно з мікро (індивідуальним), запропонованим двома іншими вимірами. Починаємо з загального огляду, звертаючись до загальної карти абстракцій, пропонуваної тим, хто хоче запрограмувати організацію в мультиагентній системі. Важливість цього виміру підкреслюється з точки зору мультиагентного програмування, розробленого в цій дисципліні. Як тільки всі концепції та абстракції будуть пояснені та розміщені відносно один одного, ми обговорюємо модель виконання організації, в якій вводяться подальші абстракції, пов'язані з організацією. Ми закінчуємо лекцію примітками та історичними

		<i>фактами, щоб зацікавлений слухач міг отримати глибші знання щодо підходів, прямо чи опосередковано пов'язаних з організаціями та їх використанням у сфері мультиагентної системи.</i>
8	<i>Лекція 7. Програмування організації ситуативних агентів</i>	<i>У цій лекції ми бачимо організаційний вимір на практиці в подальшому розширенні сценарію розумної кімнати, залучаючи більш чіткі та складні координаційні схеми між агентами. Ми переробляємо сценарій на основі організаційних концепцій та абстракцій програмування, представлених у попередній лекції, обговорюючи, зокрема, як можна змінити схеми координації між агентами для досягнення більшої гнучкості та відкритості. Ця лекція також досліджує, як агентів можна запрограмувати міркувати про організацію, в якій вони беруть участь, щоб, наприклад, вони могли виявити певну роль, що представляє інтерес з точки зору місії, цілей та обов'язків.</i>
9	<i>Лекція 8. Інтеграція з іншими технологіями</i>	<i>Практичним аспектом, важливим для вирішення питань інженерії реальних систем, є те, як MAOP та допоміжні технології - JaCaMo - в даному випадку - можуть бути інтегровані з існуючими бібліотеками та технологіями, які базуються на інших парадигмах. Окрім впровадження технічних мостів, це також вимагає розуміння інтеграції з більш концептуальної точки зору. Інтеграція може здійснюватися у двох напрямках: або вбудовувати всередину програми MAS якусь існуючу технологію, таку як бібліотека чи фреймворк, або інтегрувати MAS всередину якоїсь існуючої платформи. У цій лекції обговорюємо ці аспекти, спочатку надаючи вказівки щодо інтеграції залежно від конкретної технології, з якою слід інтегрувати, а потім обговорюючи деякі випадки в конкретних доменах додатків.</i>
10	<i>Лекція 9. Підведення підсумків та перспективи</i>	<i>У цій заключній лекції ми подаємо огляд деяких основних перспектив дослідження, які стосуються MAOP. Згадавши ключові моменти, що характеризують підхід MAOP, ми зосереджуємось на ШІ та обговорюємо, як класичні проблеми, пов'язані з ШІ, можна поставити по-іншому, розглядаючи перспективу MAOP. Потім ми переходимо до програмної інженерії, обговорюючи, як технічні засоби для проектування складних програмних систем можуть підтримуватися MAOP та, загальніше, агентно-орієнтований підхід до програмної інженерії. Протягом усієї лекції пропонується посилання на вже виконані та опубліковані дослідницькі роботи, а також окреслюються майбутні напрямки досліджень.</i>

6. Самостійна робота студента/аспіранта

Дисципліна «МУЛЬТИАГЕНТНЕ ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» ґрунтується на самостійній підготовці до аудиторних занять на теоретичні та практичні теми.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

- Відвідування занять з комп'ютерного практикуму може бути епізодичним та за потреби захисту робіт комп'ютерного практикуму.
- Правила поведінки на заняттях: активність, повага до присутніх, відключення телефонів.
- Дотримання політики академічної доброчесності.
- Правила призначення заохочувальних та штрафних балів є наступними.

Заохочувальні бали нараховуються за:

- точні та повні відповіді під час опитувань за матеріалами лекцій. Протягом семестру на лекціях відбувається **бліц-опитування** за темами минулих лекцій. Максимальна кількість балів за блиц-опитування: 3 бали.
- творчий підхід у виконанні робіт комп'ютерного практикуму. Максимальна кількість балів за всі роботи – 2 бали.

Штрафні бали нараховуються за:

- плагіат (код програми не відповідає варіанту завдання, ідентичність коду програми серед різних робіт) у роботах комп'ютерного практикуму: -5 балів за кожну спробу.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Протягом семестру аспіранти виконують 8 комп'ютерних практикумів. Максимальна кількість балів за кожний комп'ютерний практикум: 6 балів.

Бали нараховуються за:

- якість виконання комп'ютерного практикуму: 0-2 бали;
- відповідь під час захисту комп'ютерного практикуму: 0-2 бали;
- своєчасне представлення роботи до захисту: 0-2 бали.

Критерії оцінювання якості виконання:

- 2 бали – робота виконана якісно, в повному обсязі;
- 1 бал – робота виконана в повному обсязі, але містить незначні помилки;
- 0 балів – робота виконана не в повному обсязі, або містить суттєві помилки.

Критерії оцінювання відповіді:

- 2 бали – відповідь повна, добре аргументована;
- 1 бал – в цілому відповідь правильна, але має недоліки або незначні помилки;
- 0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання своєчасності представлення роботи до захисту:

- 2 бали – робота представлена до захисту не пізніше вказаного терміну;
- 0 балів – робота представлена до захисту пізніше вказаного терміну.

Максимальна кількість балів за виконання та захист комп'ютерних практикумів:
6 балів × 8 комп. практ. = 48 балів.

Протягом семестру на лекціях відбувається **опитування за темою поточного заняття**. Максимальна кількість балів за опитування, яку можна отримати протягом семестру: 2 бали.

Завдання на **модульну контрольну роботу** складається з 3 запитань – 2 теоретичних та 1 практичного. Відповідь на кожне теоретичне запитання оцінюється 15 балами, а відповідь на практичне запитання оцінюється 20 балами.

Критерії оцінювання кожного теоретичного запитання контрольної роботи:

14-15 балів – відповідь правильна, повна, добре аргументована;

11-13 балів – відповідь правильна, розгорнута, але не дуже добре аргументована;

8-10 балів – в цілому відповідь правильна, але має недоліки;

5-7 балів – у відповіді є незначні помилки;

1-4 бали – у відповіді є суттєві помилки;

0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання практичного запитання контрольної роботи:

18-20 балів – відповідь правильна, розрахунки виконані у повному обсязі;

14-17 балів – відповідь правильна, але не дуже добре підкріплена розрахунками;

9-13 балів – в цілому відповідь правильна, але має недоліки;

5-8 балів – у відповіді є незначні помилки;

1-4 бали – у відповіді є суттєві помилки;

0 балів – немає відповіді або відповідь неправильна.

Максимальна кількість балів за модульну контрольну роботу:

15 балів × 2 теоретичні запитання + 20 балів × 1 практичне запитання = 50 балів.

Рейтингова шкала з дисципліни дорівнює:

$R_c = R_{\text{ком.практ}} + R_{\text{опитув}} + R_{\text{МКР}} = 48 \text{ балів} + 2 \text{ бали} + 50 \text{ балів} = 100 \text{ балів}.$

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

На першій атестації (8-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів, яку може отримати студент до першої атестації.

На другій атестації (14-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів, яку може отримати студент до другої атестації.

Семестровий контроль: **залік**

Умови допуску до семестрового контролю:

За семестрового рейтингу (R_c) не менше 60 балів та зарахуванні усіх робіт комп'ютерного практикуму аспірант отримує залік «автоматом» відповідно до таблиці (Таблиця відповідності рейтингових балів оцінкам за університетською шкалою). В іншому разі він має виконати залікову контрольну роботу.

Необхідною умовою допуску до виконання залікової контрольної роботи є виконання і захист комп'ютерного практикуму.

Аспірант може спробувати підвищити свою оцінку шляхом написання залікової контрольної роботи, при цьому його бали, отримані за семестр, анулюються («жорстка» система оцінювання).

Склад та критерії оцінювання залікової контрольної роботи:

Завдання на **залікову контрольну роботу** складається з 7 запитань – 5 теоретичних та 2 практичних. Відповідь на кожне теоретичне запитання оцінюється 10 балами, а відповідь на практичне запитання оцінюється 25 балами.

Критерії оцінювання кожного теоретичного запитання контрольної роботи:

9-10 балів – відповідь правильна, повна, добре аргументована;

7-8 балів – відповідь правильна, розгорнута, але не дуже добре аргументована;

5-6 балів – в цілому відповідь правильна, але має недоліки;

3-4 балів – у відповіді є незначні помилки;

1-2 бали – у відповіді є суттєві помилки;

0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання практичного запитання контрольної роботи:

24-25 балів – відповідь правильна, розрахунки виконані у повному обсязі;

21-23 бали – відповідь правильна, але не дуже добре підкріплена розрахунками;

17-20 балів – в цілому відповідь правильна, але має недоліки;

12-16 балів – у відповіді є незначні помилки;

1-11 бали – у відповіді є суттєві помилки;

0 балів – немає відповіді або відповідь неправильна.

Максимальна кількість балів за модульну контрольну роботу:

10 балів × 5 теоретичних запитань + 25 балів × 2 практичних запитань = 100 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Додаткова інформація з дисципліни (освітнього компонента)

Перелік запитань, які виносяться на семестровий контроль, наведено у Додатку 1.

Робочу програму навчальної дисципліни (силабус):

Складено к.т.н., доц., Баклан І.В.

Ухвалено кафедрою ІПІ ФІОТ (протокол № 16 від 29.05.2024)

Погоджено Методичною комісією факультету (протокол № 10 від 21.06.2024)