

**ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«ПРИДНІПРОВСЬКА ДЕРЖАВНА АКАДЕМІЯ БУДІВНИЦТВА ТА АРХІТЕКТУРИ»**

КАФЕДРА ПРИКЛАДНОЇ МАТЕМАТИКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
(повна назва кафедри)

ЗАТВЕРДЖУЮ»
Проректор з науково-педагогічної та навчальної роботи
В. Б. Папірник
« 15 » _____ 2019 року



СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Крос-платформне програмування»

(назва навчальної дисципліни)

спеціальність _____ 122 «Комп'ютерні науки» _____
(шифр і назва спеціальності)

освітньо-професійна програма _____ «Комп'ютерні науки» _____
(назва освітньої програми)

освітній ступінь _____ бакалавр _____
(назва освітнього ступеня)

форма навчання _____ денна _____
(денна, заочна, вечірня)

розробник _____ Шибко Оксана Миколаївна _____
(прізвище, ім'я, по батькові)

1. АНОТАЦІЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Дисципліна «Крос-платформне програмування» належить до переліку нормативних навчальних дисциплін, забезпечує професійний розвиток бакалавра та спрямована на формування в майбутнього фахівця основних понять. Крос-платформність – здатність програмного забезпечення працювати більш, ніж на одній апаратній платформі і (або) операційній системі. Забезпечується завдяки використанню високорівневих мов програмування, середовищ розробки і виконання, що підтримують умовну компіляцію, компоновку і виконання коду для різних платформ.

Крос-платформними можна назвати більшість сучасних високорівневих мов програмування. Наприклад, Java – кросплатформена мова на рівні компіляції, тобто для неї є компілятори під різні платформи. Це дозволяє – при належній якості коду – не переписувати основний двигун програми, змінюються лише особливі системозалежні частини

2. ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

	Години	Кре- ди- ти	С е м е с т р	
			V	
Всього годин за навчальним планом, з них:	105	3,5	105	
Аудиторні заняття, у т.ч:	46	1,5	46	
лекцій	30		30	
лабораторні роботи	8		8	
практичні заняття	8		8	
Самостійна робота, у т.ч:	59	1	59	
підготовка до аудиторних занять	10		10	
підготовка до контрольних заходів	10		10	
виконання курсового проекту або роботи	-			
опрацювання розділів програми, які не викладаються на лекціях	9		9	
підготовка до екзамену	30		30	
Форма підсумкового контролю			екзамен	

3. СТИСЛИЙ ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета дисципліни. Метою викладання навчальної дисципліни «Крос-платформне програмування» є навчити студентів засвоєнню необхідних знань з основ розробки крос-платформних компонентів, а також формування твердих практичних навичок щодо розроблення додатків з використанням компонентного підходу при розробки розподілених систем.

Завдання дисципліни. Основними завданнями вивчення дисципліни «Крос-платформне програмування» є ознайомлення студентів з Java – технологіями, набуття навичок застосування Java – технологій для створення крос-платформних програмних систем, умінь застосовувати створені за допомогою Java – технологій програмні системи для автоматизації професійної діяльності.

Пререквізити дисципліни. Успішне опанування курсу «Крос-платформне програмування» передбачає знання та навички з курсів «Алгоритмізація та програмування», «Об'єктно-орієнтоване програмування».

Постреквізити дисципліни. Вивчення дисципліни забезпечує досліджувати засоби багатопоточності в Java, досліджувати програмні інтерфейси виклику віддалених методів передачу віддаленого об'єкту, створювати крос-платформні програмні системи в Java, застосовувати створені за допомогою Java - технологій програмні системи для автоматизації професійної діяльності.

Компетентності:

- ІК здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у певній галузі професійної діяльності або у процесі навчання, що передбачає застосування певних теорій та методів відповідної науки і характеризується комплексністю та невизначеністю умов;

ЗК-8. Здатність генерувати нові ідеї (креативність).

ЗК-13. Здатність оцінювати та забезпечувати якість виконуваних робіт.

- СК-1 здатність до математичного та логічного мислення, формулювання та досліджування математичних моделей, зокрема дискретних математичних моделей, обґрунтування вибору методів і підходів для розв'язування теоретичних і прикладних задач в галузі комп'ютерних наук, інтерпретування отриманих результатів;

- СК-3 здатність до побудови логічних висновків, використання формальних мов і моделей

алгоритмічних обчислень, проектування, розроблення та аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем;

- СК-8 здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: структурного, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами та алгоритмами обчислень, структурами даних і механізмами управління.

Заплановані результати навчання:

- РН1 здобувати систематичні знання в галузі комп'ютерних наук, аналізувати проблеми з точки зору сучасних наукових парадигм, осмислювати і робити обґрунтовані висновки з наукової і навчальної літератури та результатів експериментів;

- РН9 застосовувати у роботі міжнародні стандарти з оцінки якості програмного забезпечення, управління та обслуговування ІТ сервісів, моделі оцінки зрілості процесів розробки ПЗ;

- РН14 використовувати формальні моделі алгоритмів та обчислюваних функцій, встановлювати розв'язність, часткову розв'язність та нерозв'язність алгоритмічних проблем, проектувати, розробляти та аналізувати алгоритми, оцінювання їх ефективності та складності;

- РН19 визначати складові структурної та параметричної ідентифікації моделей реальних систем, застосовувати методи моделювання складних об'єктів і систем з використанням відповідне програмне забезпечення, оцінювати ступінь повноти, адекватності, істинності та реалізуємості моделей реальних систем;

- РН-20 розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук, створювати надійне та ефективне програмне забезпечення.

Методи навчання – практичний, наочний, словесний, робота з книгою. Практичних навичок студенти набувають на аудиторних лекційних та практичних заняттях.

Форми навчання – аудиторна, позааудиторна, індивідуальна.

4. СТРУКТУРА (ТЕМАТИЧНИЙ ПЛАН) ДИСЦИПЛІНИ

Назва змістових модулів і тем	Кількість годин, у тому числі				
	усього	л	п	лаб	с/р
Змістовий модуль 1. Базові оператори та приведення їх типів.					
Реалізація процесів обробки інформації засобами алгоритмічних мов високого рівня.	7	4			3
Змінні та загальні типи даних.	7	2		2	3
Базові оператори та приведення типів	7	2		2	3
Управляючі інструкції	9	4		2	3
Оператори циклів	9	4		2	3
Разом за змістовим модулем 1	39	16		8	15
Змістовий модуль 2. Робота з масивами та об'єктами.					
Створення та робота з масивами	15	6	4		5

Тонкощі роботи з об'єктами	11	4	2		5
Основи бібліотеки Swing.	10	4	2		4
Разом за змістовим модулем 2	36	14	8		14
Підготовка до екзамену	30				30
Усього годин	105	30	8	8	59

5. ЛЕКЦІЙНИЙ КУРС

№ зан.	Тема занять	Кількість годин
Змістовий модуль 1.		
1,2	Реалізація процесів обробки інформації засобами алгоритмічних мов високого рівня. Поняття про автоматизацію програмування. Мова програмування Java. Структура програмного модуля.	4
3	Змінні та загальні типи даних. Змінні базових типів. Оголошення змінних. Класи-оболонки.	2
4	Базові оператори та приведення типів. Арифметичні, логічні, побітові оператори та оператор присвоювання.	2
5,6	Управляючі інструкції. Умовний оператор If. Оператор вибору switch-case. Програмування алгоритмів розгалуженої структури в середовищі Java або NetBeans.	4
7,8	Оператори циклів. Оператор For. Оператор While та Do-While Програмування алгоритмів циклічної структури в середовищі Java або NetBeans.	4
Змістовий модуль 2. Робота з масивами та об'єктами.		
9-11	Створення та робота з масивами. Одномірні та двомірні масиви. Ініціалізація, присвоювання та зрівняння масивів в середовищі Java або NetBeans.	6
12,13	Тонкощі роботи з об'єктами. Статичні поля та методи, анонімні об'єкти, внутрішні класи.	4
14,15	Основи бібліотеки Swing. Основі поняття. Створення вікон з: текстовою міткою, текстом та піктограмою, міткою та кнопкою.	4

6. ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ

№ зан.	Тема занять	Тривалість (годин)
1	Програмування в середовищі Java або NetBeans. Перша дуже проста програма.	2
2	Програмування в середовищі Java або NetBeans: Робота з датою і часом.	2
3	Створення вікон з текстовим полем в середовищі Java або NetBeans.	2
4	Основні властивості аплетів, передача аплету аргументів, аплет з елементами управління в середовищі Java або NetBeans	2

7. ТЕМИ ЛАБОРАТОРНИХ ЗАНЯТЬ

№ зан.	Тема занять	Тривалість (годин)
1	Програмування в середовищі Java або NetBeans алгоритмів типових обчислювальних процесів лінійної структури.	2
2	Програмування в середовищі Java або NetBeans алгоритмів типових обчислювальних процесів розгалуженої структури.	2
3	Програмування в середовищі Java або NetBeans алгоритмів типових обчислювальних процесів циклічної структури.	2
4	Програмування в середовищі Java або NetBeans програм обробки масивів.	2

8. САМОСТІЙНА РОБОТА

№ п/п	Назва теми	Кількість годин
1	підготовка до аудиторних занять	10
2	підготовка до контрольних заходів	10
3	виконання курсового проекту або роботи	-
4	опрацювання розділів програми, які не викладаються на лекціях:	5
	методи створення компонентів	4
	розробка та збирання компонентів	30
5	підготовка до екзамену	

9. МЕТОДИ КОНТРОЛЮ

Письмовий, усний, практична перевірка.

10. ПОРЯДОК ТА КРИТЕРІЇ ОЦІНЮВАННЯ

Контроль успішності студента здійснюється за допомогою 100-бальної системи оцінювання, що має відповідні оцінки в національній шкалі і шкалі ECTS.

Змістовий модуль 1. Базові оператори та приведення їх типів.

Самостійна домашня робота (максимальна кількість балів – 100):

«Домашнє завдання №1»

- виконання самостійної роботи та її оформлення – 60 балів;
- відповідь на теоретичне питання №1, №2 при захисті самостійної роботи (максимальна кількість балів на одне питання - 20 балів): 40 балів;
- o якщо продемонстровано високий рівень володіння матеріалом (складено код програми, отримані результати обчислення) 20 балів;
- o якщо продемонстровано добрий рівень володіння матеріалом (складено код програми з незначними помилками, отримані результати обчислення), але студент показав логічне мислення, вміння орієнтуватися в суті питання 16 – 19 балів;
- o якщо продемонстровано добрий рівень володіння матеріалом (сутність розкрита, але в кодї програми були суттєві помилки, які вплинули на результати обчислення) 6 -15 балів;
- o якщо продемонстровано низький рівень володіння матеріалом (складено код програми зі значними помилками, не отримано результатів обчислення) 1 - 5 балів;
- o якщо продемонстровано незадовільний рівень володіння матеріалом (не складено код програми, немає результатів обчислення) 0 балів.

Підсумкова оцінка зі змістового модуля 1 визначається як оцінка за самостійну домашню роботу змістового модуля 1.

Змістовий модуль 2. Робота з масивами та об'єктами**Самостійна домашня робота** (максимальна кількість балів – **100**):

«Домашнє завдання №2»

- виконання самостійної роботи та її оформлення – 60 балів;
- відповідь на теоретичне питання №1, №2 при захисті самостійної роботи (максимальна кількість балів на одне питання - 20 балів): 40 балів;
- якщо продемонстровано високий рівень володіння матеріалом (складено код програми, отримані результати обчислення) 20 балів;
- якщо продемонстровано добрий рівень володіння матеріалом (складено код програми з незначними помилками, отримані результати обчислення), але студент показав логічне мислення, вміння орієнтуватися в суті питання 16 – 19 балів;
- якщо продемонстровано добрий рівень володіння матеріалом (сутність розкрита, але в коді програми були суттєві помилки, які вплинули на результати обчислення) 6 -15 балів;
- якщо продемонстровано низький рівень володіння матеріалом (складено код програми зі значними помилками, не отримано результатів обчислення) 1 - 5 балів;
- якщо продемонстровано незадовільний рівень володіння матеріалом (не складено код програми, немає результатів обчислення) 0 балів

Підсумкова оцінка зі змістового модуля 2 визначається як оцінка за самостійну домашню роботу змістового модуля 2.

- екзамену

Екзамен проводиться в письмовій формі у вигляді відповідей на білети, що містять два теоретичних питання і 1 практичного завдання. До складання екзамену допускаються студенти, які повністю виконали навчальний план дисципліни – отримали не менше 60 балів з кожного змістового модуля.

Екзамен (максимальна кількість балів – **100**):

- відповідь на теоретичні питання (максимальна кількість балів на одне питання - 25 балів): 50 балів;
- правильна відповідь на питання 25 балів;
- відповідь на питання потребує деяких уточнень 19 – 24 балів;
- відповідь на питання із незначними помилками (сутність розкрита, але були помилки в формулюваннях) 9 -18 балів;
- відповідь на питання зі значними помилками (не розкрита сутність питання) 1 - 8 балів;
- неправильна відповідь або немає відповіді 0 балів.
- виконання практичних завдань максимальна кількість балів: 50 балів;
- правильне виконання завдання, розрахунки виконані без помилок, проведено аналіз результатів 50 балів;
- робота виконана повністю, розрахунки виконані послідовно, але виконання завдання потребує деяких уточнень, аналіз результатів недостатній або відсутній 36 – 49 балів;
- виконання завдання із незначними помилками (сутність розкрита, але були помилки в формулюваннях або розрахунках, аналіз результатів відсутній) 19 -35 балів;
- виконання завдання зі значними помилками (неправильні розрахункові формули або їх застосування, неповне або неправильне обґрунтування розв'язання завдання,) 1 - 18 балів;
- неправильне виконання завдання або немає розв'язання 0 балів.

Підсумкова оцінка з дисципліни визначається як середня між підсумковими оцінками за змістові модулі 1, 2 та оцінкою екзамену.

Порядок зарахування пропущених занять. Пропущені лекції та практичні заняття, незалежно від причини пропуску, студент відпрацьовує викладачеві (реферат, презентація) згідно з графіком консультацій

11. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Васильев А.Н. Java. Объектно-ориентированное программирование: Учебное пособие.- СПб.: Питер, 2014. – 400 с.
2. Васильев А.Н. Самоучитель Java с примерами и программами. 2-е издание. – СПб.: Наука и Техника, 2016. – 368 с.
3. Монахов В. Язык программирования Java и среда NetBeans. СПб.: БХВ, 2015. – 720 с.
4. Фишер Т.Р. Java. Карманный справочник. М.: Вильямс, 2016. 224 с.
5. Эккель Б. Философия Java. СПб.: Питер., 2014. – 640 с.
6. Хабибуллин И. Создание распределенных приложений на Java 2 / И. Хабибуллин. - М.: БХВ-Петербург, 2017. - 704 с .

Допоміжна

1. Блох Дж. Java. Эффективное программирование –М., Лори, 2014г.
2. Лафоре Р. Структура данных и алгоритмы Java. – СПб.: Питер, 2013. – 704 с.
3. Дейтел Х.М., Сантри С.И. Технология программирования на JAVA 2.-М. ООО «Бином-Приесс», 2015. – 464 с.
4. Fabio, Luigi Bellifemine Developing Multi-Agent Systems with JADE / Fabio Luigi Bellifemine. - Москва: ИЛ, 2016. - 300 с.
5. Герберт, Шилдт Java 8. Руководство для начинающих / Шилдт Герберт. - М.: Диалектика / Вильямс, 2015. – 899 с.
6. Давыдов, Станислав IntelliJ IDEA. Профессиональное программирование на Java / Станислав Давыдов , Алексей Ефимов. - М.: БХВ-Петербург, 2015. - 800 с.
7. Монахов, В. Язык программирования Java и среда NetBeans (+ DVD-ROM) / В. Монахов. - М.: БХВ-Петербург, 2017. - 704 с.
8. Сеттер, Р. В. Изучаем Java на примерах и задачах / Р.В. Сеттер. - М.: Наука и техника, 2016. - 240 с.
9. Уоллс, Крейг Spring в действии / Крейг Уоллс. - М.: ДМК Пресс, 2015. - 754 с.
10. Хабибуллин, И. Создание распределенных приложений на Java 2/ И. Хабибуллин. - М.: БХВ-Петербург, 2017. - 704 с.
11. Хеффельфингер, Дэвид Java EE 7 и сервер приложений GlassFish 4 / Дэвид Хеффельфингер. - М.: ДМК Пресс, 2016. - 332 с.
12. Язык программирования Java SE 8. Подробное описание. - М.: Вильямс, 2015. - 672 с.

12. INTERNET-РЕСУРСИ

1. <http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>
2. <http://forcoder.ru/java/struktury-dannyh-i-algoritmy-java-1386>
3. <http://forcoder.ru/java/java-effektivnoe-programirovanie-1241>

Розробник _____ (О. М. Шибко)

(підпис)

Гарант освітньої програми _____ (Н.М.Єршова)

(підпис)

Силабус затверджено на засіданні кафедри комп'ютерних наук, інформаційних технологій та прикладної математики