

**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
Р. Б. Папірник



2019 року

**СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«Конструювання програмного забезпечення»**

(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 «Конструювання програмного забезпечення» належить до переліку варіативних навчальних дисциплін, забезпечує професійний розвиток бакалавра та спрямована на формування в майбутнього фахівця основних понять. Конструювання – етап життєвого циклу програмного забезпечення (ПЗ). Конструювання ПЗ – створення ПЗ з конструкторів (блоків, операторів, функцій) і його перевірка методами верифікації і тестування. До інструментів конструювання ПЗ віднесені мови конструювання, програмні методи й інструментальні системи. Базову складову професійної діяльності фахівців в галузі програмної інженерії формують вміння та навички конструювання програмного забезпечення. До складу обов'язкового обсягу практичних навичок фахівця повинні входити поняття про методи ефективного та оптимального кодування алгоритмів.

Програміст повинен генерувати не просто будь-який код, який працює, а і обов'язково володіти якісним стилем програмування, методами документування, застосовувати методи мінімізації коду, проводити ефективний пошук помилок, зокрема не явних на етапі відладки та вміти якісно тестувати власний програмний продукт.

2. ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

	Години	Кредити	Семестр	
				IV
Всього годин за навчальним планом, з них:	120	4		120
Аудиторні заняття, у т.ч:	60	2		60
лекції	30			30
лабораторні роботи	16			16
практичні заняття	14			14
Самостійна робота, у т.ч:	60	2		60
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	10			10
підготовка до екзамену	30			30
Форма підсумкового контролю				екзамен

3. СТИСЛИЙ ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета дисципліни. Метою вивчення курсу «Конструювання програмного забезпечення» є формування бази теоретичних знань та умінь щодо сучасних методів та засобів конструювання програмних систем.

Завдання дисципліни. Завдання дисципліни «Конструювання програмного забезпечення» - надати студентам знання в сфері реалізації задач автоматизації обробки інформації, автоматизації керування об'єктами, в тому числі, сільськогосподарськими, за допомогою комп'ютерної техніки. Такі знання майбутній спеціаліст зможе застосовувати як при подальшому навчанні, так і після отримання вищої освіти у своїй професійній діяльності.

Пререквізити дисципліни. Успішне опанування курсу «Конструювання програмного забезпечення» передбачає знання та навички з курсів «Алгоритмізація та програмування», «Дискретна математика», «Дискретні структури», «Операційні системи», «Бази даних», «Об'єктно-орієнтоване програмування», «Архітектура комп'ютера».

Постреквізити дисципліни. Вміння застосовувати знання у процесі розв'язання професійних задач теоретичного та прикладного характеру при вивченні дисциплін: «Програмування мобільних пристроїв», «Моделювання програмного забезпечення», «Емпіричні методи програмної інженерії».

Компетентності:

- **ІК.** Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів комп'ютерних наук, інформаційних технологій і характеризується комплексністю та невизначеністю умов.

ЗК-1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК-2. Здатність застосовувати знання у практичних ситуаціях.

ЗК-6. Здатність вчитися і оволодівати сучасними знаннями.

ЗК-7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

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 програмні результати навчання:

PH-1. 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.

PH-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.

PH-27. Використовувати технології проектування складних систем, вибирати CASE-засоби; формулювати техніко-економічні вимоги, розробляти інформаційні та програмні системи з використанням шаблонів та засобів автоматизованого проектування.

Методи навчання – практичний, наочний, словесний, робота з книгою.

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. Загальні відомості про конструювання програмного забезпечення.					
Вступ до дисципліни «Конструювання програмного забезпечення»	4	2			2
Введення в конструювання програмного забезпечення.	12	4	2	2	4
Загальні підходи до проектування програмних систем	12	4	2	2	4
Техніка моделювання архітектури ПЗ.	12	4	2	2	4
Концепції сервіс-орієнтованої архітектури ПЗ.	12	4	2	2	4
Разом за змістовим модулем 1	52	18	8	8	18
Змістовий модуль 2. Керування конструюванням програмного забезпечення					
Методи об'єктного аналізу і моделювання	12	4	2	2	4
Рівнева організація прикладу.	8	2	2	2	2
Інтерфейси, взаємодія, еволюція програм і даних	12	4	2	2	4
Моделі якості та надійності програмних систем	6	2		2	2
Разом за змістовим модулем 2	38	12	6	8	12
Підготовка до екзамену	30				30
Усього годин	120	30	14	16	60

5. ЛЕКЦІЙНИЙ КУРС

№ зан.	Тема занять	Кількість годин
1	<p>Вступ до дисципліни «Конструювання програмного забезпечення» Мета та завдання дисципліни. Значення та місце дисципліни в системі підготовки спеціалістів комп'ютерних наук. Загальні відомості про дисципліну, її зв'язок з іншими дисциплінами. Роль комп'ютерної техніки в сучасному житті.</p>	2
2-3	<p>Введення в конструювання програмного забезпечення. Історія розвитку. Базові терміни. Елементи конструювання ПЗ. Ключові принципи конструювання. Інструменти конструювання. Мови опису, Представлення, Архітектурні каркаси. Приклади архітектурних стилів</p>	4
4-5	<p>Загальні підходи до проектування програмних систем. Моделі як методика проектування об'єктно-орієнтованих програмних систем. Моделі класів. Моделі станів. Моделі взаємодій.</p>	4
6-7	<p>Техніка моделювання архітектури. Ідентифікація цілей та ключових сценаріїв. Огляд програми. Ідентифікація ключових проблем. Вибір рішення — кандидату.</p>	4
8-9	<p>Концепції сервіс-орієнтованої архітектури ПЗ. Концепції сервіс-орієнтованої архітектури. Визначення. Концепції. Вигоди використання. Стандарти. Роль XML. Етапи циклу життя SOA. Архітектурний шаблон SOA.</p>	4
10-11	<p>Методи об'єктного аналізу і моделювання. Огляд об'єктно-орієнтованих методів аналізу і побудови моделей. Проектування архітектури програмних систем</p>	4
12	<p>Рівнева організація прикладу. Пошарова організація прикладу та виділення рівнів Представлення, Бізнес-логіки та Даних. Дизайн рівневої структури. Визначення необхідних рівнів, прийняття рішення про розподіл компонентів, визначення правил взаємодії між рівнями, визначення інтерфейсу між рівнями, вибір стратегій реалізації, Вибір протоколів взаємодії.</p>	2
13-14	<p>Інтерфейси, взаємодія, еволюція програм і даних. Визначення інтерфейсу у програмуванні. Інтерфейси в сучасних середовищах. Інтерфейс між клієнтом і сервером. Інтерфейс мов програмування. Інтерфейс і взаємозв'язок мов програмування. Взаємодія різномовних програм. Реінжиніринг програмних систем.</p>	4
15	<p>Моделі якості та надійності програмних систем. Модель якості програмних систем. Стандартні показники якості. Метрики якості. Стандартна оцінка показників якості. Керування якістю програмних систем. Моделі оцінки надійності програмних систем. Класифікація моделей надійності. Сертифікація програмного продукту.</p>	2

6. ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ

№ зан.	Тема занять	Кількість годин
1	Типові прийоми конструювання пакетів програм складної структури	2
2	Основні підходи до інтелектуалізації інформаційних систем	2
3	Життєвий цикл програмного забезпечення	2
4	Основи стандартизації при проектуванні програмних засобів	2
5	Сертифікація програмного забезпечення	2
6	Основні поняття і показники надійності програмних засобів	2
7	Забезпечення якості та надійності в процесі розробки складних програмних засобів	2

7. ТЕМИ ЛАБОРАТОРНИХ ЗАНЯТЬ

№ зан.	Тема занять	Кількість годин
1	Розробка опису і аналіз інформаційної системи	2
2	Розробка вимог до інформаційної системи	2
3	Використання моделей проектування програмного забезпечення	2
4	Планування конструювання програмного забезпечення	2
5	Використання мов конструювання програмного забезпечення	2
6	Забезпечення якості конструювання програмного забезпечення	2
7	Застосування шаблонів проектування	2
8	Документування результатів розробки проекту програмного забезпечення	2

8. САМОСТІЙНА РОБОТА

№ п/п	Назва теми	Кількість годин
1	підготовка до аудиторних занять	10
2	підготовка до контрольних заходів	10
3	опрацювання розділів програми, які не викладаються на лекціях:	
	керування конфігурацією	6
	структура і зміст SWEBOOK	4
4	підготовка до екзамену	30

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. Загальні відомості про конструювання програмного забезпечення.

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;
- правильна відповідь на питання 20 балів;
- відповідь на питання переважно правильна, але потребує деяких уточнень щодо планування та складання моделі 16 – 19 балів;
- сутність питання розкрита, але є незначні зауваження (помилки зі складання графіку або виконання етапів конструювання) 6 -15 балів;
- сутність питання розкрита частково, відповідь зі значними помилками (неправильно формулюються етапи конструювання) 1 - 5 балів;
- неправильна відповідь або немає відповіді 0 балів.

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

Змістовий модуль 2. Керування конструюванням програмного забезпечення.**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;
- правильна відповідь на питання 20 балів;
- відповідь на питання переважно правильна, але потребує деяких уточнень щодо методу об'єктного аналізу 16 – 19 балів;
- сутність питання розкрита, але є незначні зауваження (помилки зі складання графіку або виконання етапів конструювання) 6 -15 балів;
- сутність питання розкрита частково, відповідь зі значними помилками (неправильно формулюються методи якості конструювання) 1 - 5 балів;
- неправильна відповідь або немає відповіді 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 практичне 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 загалом правильна але потребує деяких уточнень щодо методології вибору 19 – 24 балів;
- сутність питання розкрита, але були помилки в визначеннях основних понять) 9 -18 балів;
- сутність питання розкрита частково, наявні суттєві помилки

в термінах дисципліни	1 - 8 балів;
○ неправильна відповідь або немає відповіді	0 балів.
- виконання практичних завдань максимальна кількість балів:	50 балів;
○ правильне виконання завдання, вибраний метод та проведено тестування програмного забезпечення	50 балів;
○ робота виконана повністю, але виконання завдання потребує деяких уточнень, аналіз результатів недостатній	40 – 49 балів;
○ робота виконана повністю, але виконання завдання потребує деяких уточнень, аналіз результатів відсутній	30 – 39 балів
○ завдання виконано частково, має незначні помилками в конструюванні або в тестуванні	20 -29 балів;
○ завдання виконано частково, зі значними помилками (неправильно проведено конструювання або тестування, неповне обґрунтування завдання)	10-19 балів;
○ завдання виконано частково, з грубими помилками (не виконано конструювання або не проведено тестування, немає обґрунтування завдання)	9 - 1 балів
○ неправильне виконання завдання або немає розв'язання	0 балів.

Підсумкова оцінка з дисципліни розраховується як середня між підсумковою оцінкою змістових модулів 1 і 2 та оцінкою екзамену.

Порядок зарахування пропущених занять. Пропущена лекція відпрацьовується підготовкою реферату відповідно до теми пропущеного заняття та його захистом. Пропущені практичні або лабораторні заняття відпрацьовуються студентами виконанням відповідної практичної або лабораторної роботи самостійно та її захистом.

11. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Лаврищева Е. М. Методы и средства инженерии программного обеспечения [Электронный ресурс] / Е. М. Лаврищева, В. А. Петрухин // Единое окно доступа к образовательным ресурсам. – Режим доступа: http://window.edu.ru/resource/699/41699/files/lavrishcheva_petrukhin.pdf
2. Липаев В. В. Программная инженерия: Методологические основы [Электронный ресурс] / В. В. Липаев // Электронная библиотека StudFiles "Все для учебы". – Режим доступа: <http://www.studfiles.ru/dir/cat32/subj1340/file14239.html>
3. Маклаков С. В. ВРwin и ERwin: CASE-средства для разработки информационных систем [Электронный ресурс] / С. В. Маклаков // Крымская Электронная Библиотека. – Режим доступа: <http://www.libkruz.com/books/2261.html>
4. Вендров А. М. Объектно-ориентированный анализ и проектирование с использованием языка UML и Rational Rose : Практикум [Электронный ресурс] / А. М. Вендров // Сайт кафедры системного программирования ВМК МГУ.
5. Трофимов С. А. CASE-технологии: практическая работа в Rational Rose [Электронный ресурс] / С. А. Трофимов //Крымская Электронная Библиотека.
6. Вигерс К. И. Разработка требований к программному обеспечению [Электронный ресурс] / К. И. Вигерс // Портал Programmer'sKlondike. – Режим доступа: http://www.proklondike.com/books/upravlenie/vigers_razrabotka_trebovaniy.html
7. ДСТУ 3918-99 (ISO/IEC 12207:1995) Інформаційна технологія. Процеси життєвого циклу програмного забезпечення (Аналог – ГОСТ Р ИСО/МЭК 12207-99 Информационная технология. Процессы жизненного цикла программных средств) [Электронный ресурс] // Электронная библиотека ГОСТов. – Режим доступа: <http://www.it-gost.ru/content/view/73/40/>

8. Лаврищева Е.М., Грищенко В.Н. Области знаний программной инженерии – SWEBOK и подход к обучению этой дисциплине// Управляющие системы и машины.– 2005.– №1.– С.38-54.
9. Бабенко Л.П., Лаврищева К.М. Основы програмної інженерії.– Навч. посібник.–К.: Знання, 2001.– 269с.
10. Лаврищева К.М. Програмна інженерія. Електронний підручник:
11. Рекомендации по преподаванию программной инженерии и информатики в университетах.–Computing Curricula-2001: Computer Science.–Пер. с англ. –Интернет– Ун. інформац. технологій, М.: 2007.– 462с.

Допоміжна

1. M. Fowler and al., Refactoring: Improving the Design of Existing Code, Addison-Wesley, 2012.
2. Russell Gold, Thomas Hammell, Tom Snyder. Test Driven Development: A J2EE Example.- Apress, 2015.- 296 pages.
3. Амблер С. Гибкие технологии: Экстремальное программирование и унифицированный процесс разработки. Библиотека программиста.—Спб.: Питер, 2015.—412 с.
4. Бек К. Экстремальное программирование: разработка через тестирование. Библиотека программиста.—СПб.: Питер, 2013.—224 с.
5. Влссидес Джон. Применение шаблонов проектирования. Дополнительные штрихи.: Пер. с англ.— М.: Издательский дом «Вильямс», 2013. — 144 с
6. Е.Гамма, Р.Хелм, Р.Джонсон, Д. Влссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования, -ДМК Пресс, 2010,-366 с.
7. Иан Соммервилл. Инженерия программного обеспечения, 6-е издание.: Пер. с англ.—М.:Издательский дом «Вильямс», 2012.—624 с.
8. Лаврищева К.М. Програмна інженерія.–К.– 2018.–319 с.
9. Мартин Р. Быстрая разработка программ: принципы, примеры, практика. Пер. с англ.—М.: Издательский дом «Вильямс», 2014.—752 с.
10. Макконел С. Совершенный код. Мастер класс / Пер. с англ.—М.: Издательско-торговый дом «Русская редакция»; Спб.: Питер, 2015.—896 с.

12. INTERNET-РЕСУРСИ

1. <https://en.wikipedia.org/wiki/Software>
2. [https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658098\(v=pandp.10\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658098(v=pandp.10)?redirectedfrom=MSDN)
3. [https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658084\(v=pandp.10\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658084(v=pandp.10)?redirectedfrom=MSDN)
4. <http://www.programsfactory.univ.kiev.ua/ru/content/books/2>

Розробник _____ (О. М. Шибко)
(підпис)

Гарант освітньої програми _____ (Н.М.Єршова)
(підпис)