

Київський університет імені Бориса Грінченка  
Факультет інформаційних технологій та математики  
Кафедра комп'ютерних наук

«ЗАТВЕРДЖУЮ»  
Проректор з науково-методичної  
та навчальної роботи  
  
Олексій ЖИЛЬЦОВ  
« 01 » 09 2022 р.

## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**ПРОГРАМУВАННЯ: Крос-платформне програмування**

спеціальності	<u>122 Комп'ютерні науки</u>
освітнього рівня	<u>першого (бакалаврського)</u>
освітньої програми	<u>122.00.01 Інформатика</u>



Київ – 2022

**Розробники:**

Яскевич Владислав Олександрович, кандидат технічних наук, доцент кафедри комп'ютерних наук і математики

**Викладачі:**

Яскевич Владислав Олександрович, кандидат технічних наук, доцент кафедри комп'ютерних наук і математики

Робочу програму розглянуто і затверджено на засіданні кафедри комп'ютерних наук і математики

Протокол від 28.08.2020 р. № 10

Завідувач кафедри  О.С. Литвин

Робочу програму погоджено з гарантом освітньої програми Інформатика

28.08.2020 р.

Керівник освітньої програми  (І.В.Машкіна)


Робочу програму перевірено

    .     . 20     р.

Заступник директора/декана  (Л.О.Мельник)

**Пролонговано:**

на 2021/2022 н.р.  (О.М.) (Амбала В.С.) (В.С.), « 26 » 08 2021 р., протокол № 10

на 2021/2022 н.р.  (О.М.) (Майвник В.) (В.), « 08 » 08 2021 р., протокол № 10

на 20    /20     н.р.      (      ), «      »      20     р., протокол №     

на 20    /20     н.р.      (      ), «      »      20     р., протокол №

## 1. Опис навчальної дисципліни

Найменування показників	Характеристика дисципліни за формами навчання
	денна
Вид дисципліни	обов'язкова
Мова викладання, навчання та оцінювання	українська
Загальний обсяг кредитів / годин	5/150
Курс	4
Семестр	7
Кількість змістових модулів з розподілом:	4
Обсяг кредитів	5
Обсяг годин, в тому числі:	150
Аудиторні	56
Модульний контроль	8
Семестровий контроль	30
Самостійна робота	56
Форма семестрового контролю	іспит

## 2. Мета та завдання навчальної дисципліни

Робоча навчальна програма з курсу «Програмування: Крос-платформне програмування» є нормативним документом Київського університету імені Бориса Грінченка, який розроблено кафедрою комп'ютерних наук і математики на основі освітньо професійної

програми підготовки здобувачів першого (бакалаврського) рівня відповідно до навчального плану спеціальності 122 ( комп'ютерні науки та інформаційні технології) Робочу навчальну програму укладено згідно з вимогами Європейської кредитної трансферно-накопичувальної системи (ЄКТС) організації навчання.

Програма визначає обсяги знань, якими повинен опанувати здобувач першого (бакалаврського) рівня відповідно до вимог освітньо-кваліфікаційної характеристики, алгоритму вивчення навчального матеріалу дисципліни «Програмування: Крос платформне програмування» та необхідне методичне забезпечення, складові і технологію оцінювання навчальних досягнень студентів.

Навчальна дисципліна «Програмування: Крос-платформне програмування» складається з чотирьох змістових модулів Процес розробки програмного забезпечення та тестування, Тестові випадки та техніки тест дизайну, Розробка через тестування, Автоматизація тестування. Обсяг дисципліни – 150 год (5 кредитів).

**Метою** вивчення навчальної дисципліни «Програмування: Крос-платформне програмування» є формування системи понять, знань, умінь і навичок, що включає в себе опанування базових знань з сучасних технологій створення крос-платформних додатків.

Основними **завданнями** вивчення дисципліни є надання студентам необхідних знань та практичних навичок з програмування та набуття **наступних компетентностей: 1)**

### **Загальні компетентності**

Здатність до комплексного розв'язання проблем. Розуміння поставленої задачі; здатність проникати в суть явища, проблеми, завдання, виявляти характерні ознаки, суттєві риси та взаємозв'язки, проводити аналогії, узагальнювати; володіння системним, цілісним підходом до аналізу і оцінки ситуації та вирішення проблеми

Креативність. Відкритість до нових знань, ідей і технологій; здатність продукувати нестандартні ідеї, творчо підходити до вирішення проблеми чи виконання завдання.

Когнітивна гнучкість. Здатність здобувати нові знання, уміння та інтегрувати їх з уже наявними; спроможність аналізувати явище, ситуацію, проблему, враховуючи різні параметри, фактори, причини; здатність адаптувати мислення для вирішення задач в змінених умовах чи нестандартних ситуаціях.

Клієнт-орієнтованість. Здатність ефективно спілкуватись із замовником, формулювати технічне завдання, розробляти план його виконання, оцінювати та забезпечувати якість виконуваних робіт, представляти результати роботи й обґрунтовувати запропоновані рішення на сучасному науково-технічному й професійному рівні.

### **2) Фахові компетентності**

Володіння сучасними методами та технологіями проектування програм та програмних комплексів, розробки оптимальних рішень щодо складу програмного забезпечення. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу програмних систем, продуктів і сервісів відповідно до вимог замовника.

## **3. Результати навчання за дисципліною**

У результаті вивчення навчальної дисципліни студент повинен

**знати:**

- особливості різних парадигм програмування, принципів, моделей, методів і технологій проектування і розроблення програмних продуктів різного призначення;

- базові поняття теорії алгоритмів, формальних моделей алгоритмів, функцій, питань обчислюваності та розв'язності масових проблем, понять часової та просторової складності алгоритмів при розв'язанні обчислювальних задач;

- структури даних та фундаментальних алгоритмів, методології та інструментальних засобів об'єктно-орієнтованого аналізу та проектування; - технологій створення мобільних додатків, крос-платформного програмування; - стандартів, методів, технологій і засобів управління процесами життєвого

циклу програмних систем, продуктів і сервісів інформаційних технологій; **уміти:**

- використовувати формальні моделі алгоритмів та обчислюваних функцій,

встановлювати розв'язність, часткову розв'язність та нерозв'язність алгоритмічних проблем, проектувати, розробляти та аналізувати алгоритми, оцінювати їх ефективності та складності;

- 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;

- створювати мобільні додатки із застосуванням сучасних технологій та інструментальних засобів;

та досягти наступних програмних результатів навчання:

- ДПРз-1 технологій створення Веб- та мобільних додатків, крос-платформного програмування;

- ДПРу-1 створювати Веб- та мобільні додатки із застосуванням су-часних технологій та інструментальних засобів;

#### 4. Структура навчальної дисципліни

##### Тематичний план для денної форми навчання

Назви змістових модулів і тем	Кількість годин				
	денна форма				
	Усь ого	у тому числі			
л.		пр.	м.к.	с.р.	
<b>7 семестр</b>					
<b>Змістовий модуль 1. Процес розробки програмного забезпечення та тестування</b>					
Тема 1. Процес розробки програмного забезпечення та тестування.	28	4	10		14
Модульний контроль 1.	2			2	
<b>Разом за змістовим модулем 1</b>	<b>30</b>	<b>4</b>	<b>10</b>	<b>2</b>	<b>14</b>
<b>Змістовий модуль 2. Тестові випадки та техніки тест дизайну</b>					
Тема 2. Тестові випадки та техніки тест дизайну.	28	4	10		14
Модульний контроль 2.	2			2	
<b>Разом за змістовим модулем 2</b>	<b>30</b>	<b>4</b>	<b>10</b>	<b>2</b>	<b>14</b>
<b>Змістовий модуль 3. Розробка через тестування</b>					
Тема 3. Розробка через тестування	28	2	12		14
Модульний контроль 3.	2			2	

<b>Разом за змістовим модулем 3</b>	<b>30</b>	<b>2</b>	<b>12</b>	<b>2</b>	<b>14</b>
<b>Змістовий модуль 4. Автоматизація тестування</b>					
Тема 4. Автоматизація тестування.	28	2	12		14
Модульний контроль 4.	2			2	
<b>Разом за змістовим модулем 4</b>	<b>30</b>	<b>2</b>	<b>12</b>	<b>2</b>	<b>14</b>
<b>Семестровий контроль</b>	<b>30</b>				
<b>Разом за 7-й семестр</b>	<b>180</b>	<b>12</b>	<b>44</b>	<b>8</b>	<b>56</b>
<b>Усього годин</b>	<b>150</b>	<b>12</b>	<b>44</b>	<b>8</b>	<b>56</b>

## 5. Програма навчальної дисципліни

### 7 семестр

Змістовий модуль 1. Процес розробки програмного забезпечення та тестування. Тема

1. Процес розробки програмного забезпечення та тестування.

Визначення тестування та його важливість. Причини виникнення помилок в ПЗ. Баг (bug), дефект (defect), збій (error) та різниця між ними. Що може бути протестоване. ЖЦР ПЗ та помилки, що виникають на кожному з етапів. Життєвий цикл тестування (Fundamental Test Process): Test Planning, Test Design, Test Execution, Result Analysis and Reporting). Основні дії та артефакти кожного з етапів..

Змістовий модуль 2. Тестові випадки та техніки тест дизайну.

Тема 2. Тестові випадки та техніки тест дизайну.

Техніка еквівалентного розбиття (Equivalence Partitioning), техніка граничних значень (Boundary Value Analysis), техніка таблиць рішень (Decision Table Testing), техніка тестування станів та переходів (State Transition Testing), техніка тестування за сценаріями використання (Use Case Testing), доменного аналізу (Domain Analysis). Аналіз системи за об'єктами-діями над об'єктами-параметрами – значеннями параметрів. Тестова комбінаторика. Попарне тестування (Pair wise).

Змістовий модуль 3. Розробка через тестування.

Тема 3. Розробка через тестування.

Помилки в коді. Налаштування: помилки та попередження. Інструменти для розробників у Chrome (Chrome Developer Tools). Встановлення зовнішніх бібліотек за допомогою npm. Тестування. Модульні тести. Покриття тестами. Функціональне тестування. Значення тестування.

Змістовий модуль 4. Автоматизація тестування.

Тема 4. Автоматизація тестування.

Фреймворки автоматизованого тестування. Selenium. API. Створення мережових запитів. Перетворення даних. Аутентифікація. Методи HTTP. Коди відповідей HTTP.

## 6. Контроль навчальних досягнень

**6.1. Система оцінювання навчальних досягнень студентів Розподіл балів, які отримують**

## студенти

### Розподіл балів, які отримують студенти у 6-му семестрі

Поточне тестування та самостійна робота				Підсумковий тест (екзамен)	Сума
Змістовий модуль 1	Змістовий модуль 2	Змістовий модуль 3	Змістовий модуль 4	40	100
15	15	15	15		

### Розрахунок рейтингових балів за видами поточного (модульного) контролю

№ з/п	Вид діяльності студента	Макс. кількість балів за одиницю	Модуль 1		Модуль 2		Модуль 3		Модуль 4	
			Кількість одиниць до розрахунку	Макс. кількість балів за вид	Кількість одиниць до розрахунку	Кількість одиниць до розрахунку	Кількість одиниць до розрахунку	Макс. кількість балів за вид	Макс. кількість балів за вид	Макс. кількість балів за вид
1	Відвідування лекцій	1	2	2	2	2	1	1	1	1
2	Відвідування практичних занять	1	5	5	5	5	6	6	6	6
3	Виконання завдань для самостійної роботи	5	1	5	1	5	1	5	1	5
4	Робота на практичних (семінарських) заняттях	10	5	50	5	50	6	60	6	60
8	Виконання модульної контрольної роботи	25	1	25	1	25	1	25	1	25
	Макс. кількість балів за видами поточного контролю (МВ)	-	-	<b>87</b>		<b>87</b>		<b>97</b>		<b>97</b>

### Методика розрахунків модульної і семестрової оцінок студента

№ з/п	Оцінка студента	Макс. оцінка	Модуль 1	Модуль 2	Модуль 3	Модуль 4
1	Максимальна підсумкова семестрова модульна оцінка (МС)	<b>60</b>	-	-	-	-
2	Максимальні підсумкові оцінки за змістовими модулями (ММ)		<b>15</b>	<b>15</b>	<b>15</b>	<b>15</b>

3	Фактична кількість балів, отриманих студентом за видами поточного контролю (приклад) ФБ)		80	75	68	79
4	Підсумкові фактичні оцінки студента за змістовими модулями $M = \text{ФБ} / \text{МВ} * \text{ММ}$		14	13	11	14
5	Підсумкова семестрова модульна оцінка студента С = $M_1 + M_2 + M_3 + M_4$		52			
6	Екзаменаційна рейтингова оцінка студента (Е)	40	34			
7	Підсумкова семестрова рейтингова оцінка студента $P = C + E$		86			

Від 1 до 8 балів додаються до підсумкової рейтингової оцінки як заохочення за участь у конференціях, наукових гуртках, участь у олімпіадах тощо.

### 6.1. Завдання для самостійної роботи та критерії її оцінювання

№ з/п	Назва теми	Кількість годин	Бали
<b>5 семестр</b>			
<b>Змістовий модуль 1. Процес розробки програмного забезпечення та тестування.</b>		<b>14</b>	<b>5</b>
1	Процес розробки програмного забезпечення та тестування.	14	5
<b>Змістовий модуль 2. Тестові випадки та техніки тест дизайну</b>		<b>14</b>	<b>5</b>
3	Тестові випадки та техніки тест дизайну	14	5
<b>Змістовий модуль 3. Розробка через тестування</b>		<b>14</b>	<b>5</b>
3	Розробка через тестування	14	5
<b>Змістовий модуль 4. Автоматизація тестування</b>		<b>14</b>	<b>5</b>
4	Автоматизація тестування	14	5
<b>Разом за 7-й семестр</b>		<b>56</b>	<b>20</b>
<b>Разом</b>		<b>56</b>	<b>20</b>



## 6.2. Форми проведення модульного контролю та критерії оцінювання.

№ з/п	Модульний контроль	Бали
1	Модульний контроль 1 (тест)	25
2	Модульний контроль 2 (тест)	25
3	Модульний контроль 3 (тест)	25
4	Модульний контроль 4 (тест)	25

22-25 балів заслуговує студент, який виявив повне знання програмного матеріалу, успішно виконує передбачені програмою завдання, засвоїв основну літературу рекомендовану програмою, виявив систематичний характер знань з дисциплін і здатний до самостійного доповнення, виконав завдання всіх трьох рівнів.

13-21 балів заслуговує студент, що виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та майбутньої роботи за професією, вміє виконувати завдання, передбачені програмою, знайомий з основною рекомендованою літературою, виконав завдання 1-2 рівнів та частково деякі завдання третього рівнів.

До 13 балів заслуговує студент, що виявив часткове знання основного програмного матеріалу, не завжди вміє виконувати завдання, передбачені програмою, знайомий лише частково з основною рекомендованою літературою, виконав завдання першого рівня.

## 6.3. Форми проведення семестрового контролю та критерії оцінювання

**Семестровий контроль – іспит (письмово), максимальна оцінка – 40 балів**  
Екзаменаційний білет містить завдання двох рівнів: 25 тестових з відкритою відповіддю (на одну дію), 1 практичне завдання підвищеного рівня. Кожне з завдань екзаменаційного білета оцінюється за шкалою:

- тестові завдання з відкритою відповіддю – по 1 балу за кожне завдання; ● завдання підвищеного рівня – до 15 балів за завдання.

## 6.4. Орієнтовний перелік питань та задач для семестрового контролю. 1.

Що таке життєвий цикл ПЗ? Які його етапи ви знаєте?

2. Визначте місце тестування на етапах розробки програмного продукту. 3. Які переваги та недоліки класичної моделі?

4. Які переваги та недоліки інкрементної моделі?

5. Які переваги та недоліки спіральної моделі?

6. Дайте визначення терміну „якість програмного забезпечення” та поясніть його. 7. Визначення термінів “помилка”, „дефект”, „відмова” та різниця між ними. 8. Поясніть, що може бути протестоване.

9. Дайте визначення термінам “тестування ПЗ”, „відлагодження ПЗ”, у чому полягає різниця між ними.

10. Статичне тестування, динамічне тестування та різниця між ними. 11. Дайте визначення

- термінам верифікація та валідація, у чому полягає різниця між ними.
12. Мета застосування модульного тестування, інтеграційного тестування, системного тестування.
  13. Дайте визначення термінам “регресійне тестування”, „re-test” (повторне тестування) та поясніть різницю між ними.
  14. Основні дії та артефакти етапу планування процесу тестування (Test Planning).
  15. Основні дії та артефакти етапу виконання тестів (Test Execution).
  16. Рівні тестування.
  17. У чому відмінність між технікою еквівалентного розбиття та аналізу граничних значень. Приведіть практичний приклад.
  18. Техніка доменного аналізу. Поясніть її застосування на прикладі.
  19. У чому полягає статичний аналіз коду (static analysis) та які помилки за його допомогою можна виявити?
  20. Статичні техніки тестування
  21. Що таке інспекція та етапи його проведення?
  22. Що таке „тестування критичного шляху” (critical path test), „димове тестування” 23. (smoke test), розширене тестування”. У яких випадках кожен з цих видів тестування виконується.
  24. Визначення чек-листа, тест-кейса, тестового набору і різниця між ними. 25. Формат запису чек-листа.
  26. Звіт про дефект, його структура.
  27. Серйозність (severity) та важливість (priority) дефекту.
  28. Що таке розробка ПЗ через тестування?
  29. Що таке керована поведінкою розробка?
  30. Що таке принцип AAA?
  31. Опишіть API Assert бібліотеки Chai?
  32. Що таке покриття тестами?
  33. Які є фреймворки та бібліотеки для модульного тестування в JavaScript? 34. Яка структура тесту в Mocha
  35. Які принципи тестування Chai за допомогою Assert?
  36. Які принципи тестування Chai за допомогою Expect?
  37. Які принципи тестування Chai за допомогою Should?

## 6.6. Шкала відповідності оцінок

Оцінка за стобальною шкалою	Рейтингова оцінка	Значення оцінки
90-100	A	<i>Відмінно</i> – відмінний рівень знань (умінь) в межах обов’язкового матеріалу з можливими незначними недоліками
82-89	B	<i>Дуже добре</i> – достатньо високий рівень знань (умінь) в межах обов’язкового матеріалу без суттєвих грубих помилок
75-81	C	<i>Добре</i> – загалом добрий рівень знань (умінь) в межах обов’язкового матеріалу з незначною кількістю

		ПОМИЛОК
69-74	D	<i>Задовільно</i> – посередній рівень знань (умінь) в межах обов'язкового матеріалу із значною кількістю недоліків достатній для подальшого навчання або професійної діяльності
60-68	E	<i>Достатньо</i> – мінімально допустимий рівень знань (умінь) в межах обов'язкового матеріалу
35-59	FX	<i>Незадовільно з можливістю повторного складання екзамену</i> – незадовільний рівень знань
1-34	F	<i>Незадовільно з обов'язковим повторним вивченням курсу</i> – низький рівень знань

## 7. Навчально-методична картка дисципліни

### 7 семестр

Разом: 150 год., із них: лекції – 12 год., лабораторна робота – 44 год., самостійна робота – 56 год., модульний контроль – 8 год., семестровий контроль – 30 год.

Модулі	Назва модуля	Теми лекцій	Теми лабораторних робіт	Самостійна робота	Види поточного контролю
Змістовний модуль 1	<b>Процес розробки програмного забезпечення та тестування</b>	Планування та організація процесу тестування Тестування документації та вимог	Планування і організація процесу тестування. Тестування документації та вимог	<b>5 балів</b>	<b>Модульна контрольна робота 1 (25 балів)</b>
Змістовний модуль 2	<b>Тестові випадки та техніки тест дизайну</b>	Розробка тестів і тестових сценаріїв Пошук і документування дефектів	Розробка тестів і тестових сценаріїв Тестування програмного продукту Звіт про дефекти Доменне тестування Дослідницьке тестування	<b>5 балів</b>	<b>Модульна контрольна робота 2 (25 балів)</b>
Змістовний модуль 3	<b>Розробка через тестування</b>	Розробка через тестування	Створення автоматичних тестів Розробка ПЗ через тестування Керована поведінкою розробка ПЗ	<b>5 балів</b>	<b>Модульна контрольна робота 2 (25 балів)</b>
Змістовний модуль 4	<b>Автоматизація тестування</b>	Особливості тестування Веб-додатків	Тестування веб додатків	<b>5 балів</b>	<b>Модульна контрольна робота 2 (25 балів)</b>

## **8. Рекомендовані джерела**

### **Базова**

1. Ткаченко О.М. Комп'ютерне програмування на мові Java. Навчальний посібник. – К.: "Аграр Медіа" 2013. 146 с.
2. Костенко А. В., Костирко В. С., Плеша М. І. Крос-платформне програмування: навч. посіб. Львів: ВЛТЕУ, 2019. 247 с.

### **Допоміжна**

1. Glenford J. Myers, Corey Sandler, Tom Badgett. The Art of Software Testing, 3rd Edition. Wiley, 2015, 256 Pages,  
<https://onlinelibrary.wiley.com/doi/book/10.1002/9781119202486>
2. W. Lewis. Software Testing and Continuous Quality Improvement. Auerbach Publications; 3rd edition (December 22, 2008), 688 pages

### **Інформаційні ресурси**

3. Навчальний курс Основи тестування програмного забезпечення [Електронний ресурс]. Режим д  
[https://courses.prometheus.org.ua/courses/course v1:LITS+115+2017\\_T4/about /](https://courses.prometheus.org.ua/courses/course v1:LITS+115+2017_T4/about/)