

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра штучного інтелекту та програмного забезпечення



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Технологія створення програмних продуктів

рівень вищої освіти	перший (бакалаврський) рівень
галузь знань	0501 Інформатика та обчислювальна техніка
напрямок	6.050101 комп'ютерні науки
освітня програма	
спеціалізація	
вид дисципліни	за вибором
факультет	комп'ютерних наук

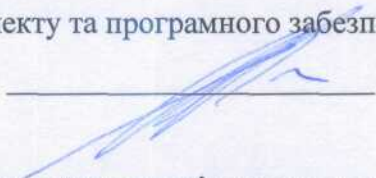
Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук
«29» серпня 2018 року, протокол № 9

РОЗРОБНИКИ ПРОГРАМИ:

Старший викладач кафедри штучного інтелекту та програмного забезпечення **Горбань
Анатолій Михайлович**


Програму схвалено на засіданні кафедри штучного інтелекту та програмного забезпечення
Протокол від «26» червня 2018 року № 11

Завідувач кафедри штучного інтелекту та програмного забезпечення


Куклін В.М.

Програму погоджено методичною комісією факультету комп'ютерних наук
Протокол від «27» червня 2018 року № 7

Голова методичної комісії факультету комп'ютерних наук


(Васильєва Л.В.)



ВСТУП

Програма навчальної дисципліни “Технологія створення програмних продуктів” складена відповідно до освітньо-професійної програми підготовки (бакалаврського) рівня вищої освіти, напрямів підготовки: 6.050101 «Комп’ютерні науки».

1. Опис навчальної дисципліни

1.1. Основною метою є надання уявлення про розробку ПС як промисловий процес, що базується на системі інженерних принципів та технологічних процесах, орієнтованих на досягнення максимальної продуктивності середніх та великих груп розробників; засвоєння теоретичних та практичних знань та навичок в області промислової розробки ПС.

1.2. Формування у студентів знань про парадигми процесів розробки програмного забезпечення, організацію процесу управління проектом, методи забезпечення та оцінки якості програм, технологічні процеси етапів аналізу, проектування, тестування та супроводження ПС.

Студенти мають оволодіти знаннями базових понять, принципів, методів та процедур прикладного системного аналізу.

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин – 120

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
4-й	
Семестр	
7-й	
Лекції	
32 год.	_____ год.
Практичні, семінарські заняття	
_____ год.	_____ год.
Лабораторні заняття	
32 год.	_____ год.
Самостійна робота	
56 год.	_____ год.
Індивідуальні завдання	
_____ год.	_____ год..

1.6. Заплановані результати навчання

знати:

- виявлення та встановлення вимог до системи;
- вибір методів та засобів аналізу систем.

вміти:

- розрахунку проектних метрик та оцінки на їх основі параметрів проекту,
- планування типових проектних завдань,
- структурування і модульної оптимізації ПС,
- планування і розробки тестових варіантів.

2. Тематичний план навчальної дисципліни*Розділ 1. Організація процесу розробки ПС**Тема 1. Основні поняття.*

Визначення технології розробки програмного забезпечення. Проблеми надійності ПЗ та моделі якості процесів розробки.

Тема 2. Парадигми розробки

Парадигми класичного життєвого циклу та макетування.

Тема 3. Стратегії.

Поняття стратегії розробки програмного забезпечення. Стратегії на основі класичного життєвого циклу — інкрементна модель та модель швидкої розробки (RAD).

Тема 4. Еволюційні моделі розробки

Моделі розробки на основі макетування — спіральна модель, компонентно-орієнтована модель та XP-процес

*Розділ 2. Керівництво програмними проектами**Тема 1. Процес керівництва проектом*

Завдання та роль процесу керівництва на різних етапах проекту, роль вимірювань, та метрик. Оцінка проекту, аналіз ризиків, трасування та контроль.

Тема 2. Планування

Планування проекту та планування проектних задач у відповідності до вибраної парадигми розробки.

Тема 3. LOC-метрики

Розмірно-орієнтовані метрики — їх розрахунок та виконання оцінки проекту на їх основі. Конструктивна модель вартості COSOMO та її використання.

Тема 4. FP-метрики

Функціонально-орієнтовані метрики — їх розрахунок та виконання оцінки проекту на їх основі.

*Розділ 3. Основи проектування ПС**Тема 1. Особливості етапу проектування*

Особливості процесу синтезу ПС, основні завдання підетапів: структурування системи, моделювання управління, декомпозиція підсистем.

Тема 2. Характеристика модульної структури ПС

Модульність, інформаційна закритість. Метрики зв'язності та зчеплення.

Тема 3. Класичні методи проектування

Метод структурного проектування, використання шаблонів проектування та компонентний підхід.

Тема 4. Складність ПС

Поняття складності системи, метрики складності та характеристики ієрархічної структури системи.

Розділ 4. Тестування ПС

Тема 1. Основні поняття.

Основні поняття, принципи та види тестування програмного забезпечення.

Тема 2. Структурне тестування

Особливості тестування “білого ящика”, способи тестування базового шляху, умов, потоків даних та циклів.

Тема 3. Функціональне тестування

Особливості тестування “чорного ящика”, способи розбиття по еквівалентності, аналізу граничних умов, діаграм причин-наслідків.

Тема 4. Організація процесу тестування

Методика тестування елементів ПС, інтеграції, правильності та системне тестування.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
Розділ 1. Організація процесу розробки ПС						
Тема 1. Основні поняття	6	2				4
Тема 2. Парадигми розробки	6	2				4
Тема 3. Стратегії	6	2				4
Тема 4. Еволюційні моделі розробки	6	2				4
Разом за розділом 1	24	8				16
Розділ 2. Керівництво програмними проектами						
Тема 1. Процес керівництва проектом	6	2	2			2
Тема 2. Планування	6	2	2			2
Тема 3. LOC-метрики	10	2	4			4
Тема 4. FP-метрики	10	2	4			4
Разом за розділом 2	32	8	12			12
Розділ 3. Основи проектування ПС						
Тема 1. Особливості етапу проектування	7	2	1			4
Тема 2. Характеристики модульної структури ПС	6	2	2			2
Тема 3. Класичні методи проектування	7	2	1			4
Тема 4. Складність ПС	6	2	2			2
Разом за розділом 3	26	8	6			12
Розділ 4. Тестування ПС						
Тема 1. Основні поняття	10	2	4			4
Тема 2. Структурне тестування	8	2	2			4
Тема 3. Функціональне тестування	10	2	4			4
Тема 4. Організація процесу тестування	10	2	4			4
Разом за розділом 4	38	8	14			16
Усього годин	120	32	32			56

4. Теми практичних (лабораторних) занять

№ з/п	Назва теми	Кількість годин
Розділ 2. Керівництво програмними проектами		
1	Оцінка проекту на основі LOC-оцінок	2
2	Оцінка проекту за допомогою конструктивної моделі вартості	2
3	Розрахунки показників зв'язності програм	4
4	Розрахунки показників зчеплення програм	4
Розділ 3. Основи проектування ПС		
5	Оцінки характеристик ієрархічної структури програм	6
Розділ 4. Тестування ПС		
6	Підрахунок показника цикломатичної складності для різних мов програмування	4
7	Формування тестових варіантів по способу тестування базового шляху	2
8	Формування тестових варіантів по способу тестування потоків даних	2
9	Формування тестових варіантів по способу розбиття по еквівалентності	2
10	Формування тестових варіантів по способу аналізу граничних значень	4
Разом:		32

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
Розділ 1. Організація процесу розробки ПС		
1	Знайомство з основними положеннями дисципліни - за джерелами з списку рекомендованої літератури	16
Розділ 2. Керівництво програмними проектами		
2	Вивчення методології використання моделі СОСОМО II для етапів композиції ПС, раннього етапу проектування, етапу постархітектури та аналізу чутливості проекту.	12
Розділ 3. Основи проектування ПС		
3	Напрацювання навичок оцінки метрик якості модулів на основі аналізу кодів відкритого ПЗ.	12
Розділ 4. Тестування ПС		
4	Планування тестових варіантів та планів тестування власних (в т. ч. Дипломних) проектів.	16
Разом		56

6. Індивідуальні завдання

(немає)

7. Методи контролю

На протязі усього терміну викладання означеної дисципліни проводиться поточний контроль засвоєння лекційного матеріалу (контроль знань) та контроль здобуття практичних навичок (контроль вмінь). Підсумковий семестровий контроль також дозволяє контролювати як одержані знання так і набуті вміння:

- Після закінчення викладання кожного модуля курсу контроль знань проводиться у вигляді тестування по матеріалам розділа.
- Після викладання курсу здійснюється підсумковий контроль.

8. Схема нарахування балів

Розподіл балів для підсумкового семестрового контролю в формі заліку без виконання залікової роботи

Поточний контроль та самостійна робота					
Розділ 1	Розділ 2	Розділ 3	Розділ 4	Контрольна робота, передбачена навчальним планом	Разом
20	20	20	20	20	100

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

Критерії оцінювання

Оцінка «**зараховано**» (50-100 балів) ставиться студентові, який виявив знання основного навчального матеріалу в обсязі, необхідному для подальшого навчання і майбутньої роботи за фахом, здатний виконувати завдання, передбаченні програмою, ознайомлений з основною рекомендованою літературою; при виконанні завдань допускається помилок, але демонструє спроможність їх усувати.

Оцінка «**незараховано**» (1-49 балів) ставиться студентові, який допускає принципові помилки у виконанні передбачених програмою завдань, не може продовжити навчання чи розпочати професійну діяльність без додаткових занять з відповідної дисципліни.

9. Рекомендована література

Основна література

1. И. Соммервилл. Инженерия программного обеспечения. М.: Вильямс, 2002.
2. Л. Константайн, Л. Локвуд. Разработка программного обеспечения. СПб.: Питер, 2004.
3. Канер Кем, Фолк Джек, Нгуен Енг Кек Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений. — Киев: ДиаСофт, 2001.
4. С. Орлов. Технологии разработки программного обеспечения. СПб.: Питер, 2003.

Допоміжна література

1. Л. Басс, П. Клементс, Р. Кацман. Архитектура программного обеспечения на практике. СПб.: Питер, 2006.
2. У. Ройс. Управление проектами по созданию программного обеспечения. М.: Лори, 2002.

3. Г. Буч, Дж. Рамбо, А. Джекобсон. Язык UML. Руководство пользователя. М.: ДМК, 2000.
4. Бейзер Б. Тестирование чёрного ящика. Технологии функционального тестирования программного обеспечения и систем. — СПб.: Питер, 2004.
5. Э. Дж. Брауде. Технология разработки программного обеспечения. СПб.: Питер, 2004.
6. Е. А. Жоголев. Лекции по технологии программирования: Учебное пособие. М., Издательский отдел факультета ВМиК МГУ, 2001.
7. Д. Леффингуэлл, Д. Уидриг. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: Вильямс, 2002.
8. А. Якобсон, Г. Буч, Дж. Рамбо. Унифицированный процесс разработки программного обеспечения. СПб.: Питер, 2002.
9. К. Бек. Экстремальное программирование. СПб.: Питер, 2002.
10. В. В. Липаев. Методы обеспечения качества крупномасштабных программных средств. М., Синтег, 2003.
11. Э. Хант, Д. Томас. Программист-прагматик. М.: Лори, 2004.
12. Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. СПб.: Питер-ДМК, 2001.
13. М. Фаулер и др. Архитектура корпоративных программных приложений. М.: Вильямс, 2004.
14. Б. Бозм, Дж. Браун, Х. Каспар и др. Характеристики качества программного обеспечения. М.: Мир, 1991.
15. Дж. Рамбо, А. Якобсон, Г. Буч. UML: Специальный справочник. СПб.: Питер, 2002.
16. М. Фаулер, К. Скотт. UML в кратком изложении. М., Мир, 1999.
17. Э. Таненбаум, М. ван Стеен. Распределенные системы. Принципы и парадигмы. СПб.: Питер, 2003.
18. Э. Таненбаум. Современные операционные системы. 2-е издание. СПб.: Питер, 2002.
19. Э. Таненбаум. Компьютерные сети. 4-е издание. СПб.: Питер, 2003.
20. Т. Кормен, Ч. Лейзерсон, Р. Ривест. Алгоритмы: построение и анализ. М.: МЦНМО, 1999.
21. Э. М. Кларк, О. Грамберг, Д. Пелед. Верификация моделей программ: Model Checking. М.: МЦНМО, 2002.

10. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. Software Engineering Body of Knowledge: [Електроний ресурс] // 2005.URL: http://www.swebok.org/ironman/pdf/SWEBOK_Guide_2004.pdf
2. F. Bachmann, L. Bass, C. Buhman, S. Comella-Dorda, F. Long, J. Robert, R. Seacord, K. Wallnau. Volume II: Technical Concepts of Component-Based Software Engineering.: [Електроний ресурс] // 2nd Edition/ Technical Report CMU/SEI-2000-TR-008. URL:<http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr008.pdf>
3. G. Alonso, F. Casati, H. Kuno, V. Machiraju. Web Services. Concepts, Architectures and Applications: [Електроний ресурс] // Springer-Verlag, 2004. URL: <http://www.inf.ethz.ch/personal/alonso/WebServicesBook>

4. Capability Maturity Model Integration (CMMI), Version 1.1. CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS, V1.1). Staged Representation.: [Электронный ресурс] // SEI Technical Report CMU/SEI-2002-TR-012, Software Engineering Institute, Pittsburgh, March 2002. URL: <http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr012.pdf>
5. Документация по технологиям Java: [Электронный ресурс] //URL: <http://java.sun.com/docs/index.html>
6. Документация по платформе .NET : [Электронный ресурс] // URL: <http://msdn.microsoft.com/library/default.asp>
7. Расширяемый язык разметки (XML) 1.0 (русский перевод первой версии стандарта) : [Электронный ресурс] // URL.: <http://www.rol.ru/news/it/helpdesk/xml01.htm>
8. В. В. Головач. Дизайн пользовательского интерфейса: [Электронный ресурс] // URL: <http://www.uibook1.ru>.
9. Портал специалистов по тестированию и обеспечению качества ПО: [Электронный ресурс] // URL:<http://software-testing.ru>.
10. Портал об автоматизированном тестировании ПО: [Электронный ресурс] // URL: <http://automated-testing.info>.