



Технологія створення програмних продуктів

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>16 Хімічна та біоінженерія</i>
Спеціальність	<i>163 Біомедична інженерія</i>
Освітня програма	<i>Медична інженерія, Регенеративна та біофармацевтична інженерія</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>Очна (денна)/змішана/дистанційна</i>
Рік підготовки, семестр	<i>4 курс, весняний семестр</i>
Обсяг дисципліни	<i>4 кредити ЄКТС (120 годин)</i>
Семестровий контроль/ контрольні заходи	<i>Залік, МКР, реферат</i>
Розклад занять	<i>Лекції (26 годин) – 3 заняття/ 2 тижні, практичні заняття (28 годин) – 3 заняття/ 2 тижні (Згідно розкладу на сайті http://rozklad.kpi.ua/)</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: канд. фіз.-мат. наук, доцент, Соломін Андрій Вячеславович, a.solomin@kpi.ua; andr-sol@i.ua Практичні: канд. фіз.-мат. наук, доцент, Соломін Андрій Вячеславович, a.solomin@kpi.ua; andr-sol@i.ua</i>
Профіль викладача	<i>Соломін А.В.: https://intellect.kpi.ua/profile/sav231 http://bmi.fbmi.kpi.ua/department/staff-department/</i>
Розміщення курсу	<i>https://do.ipu.kpi.ua/course/view.php?id=2282</i>

Програма навчальної дисципліни

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

Мета дисципліни

Головною метою навчальної дисципліни «Технологія створення програмних продуктів» є формування у студентів здатностей проектувати, конструювати, тестувати, налаштовувати, впроваджувати та супроводжувати програмні системи, обираючи оптимальні рішення на базі сучасних технологій; розробляти компоненти програмного забезпечення; реалізовувати прототипи архітектури програмного забезпечення; вільно орієнтуватися в сучасних тенденціях розвитку галузі; застосовувати спеціальні знання в галузі інформатики та обчислювальної техніки для розв'язання міждисциплінарних інженерних задач, а також ефективно працювати при розв'язанні інноваційних інженерних задач.

Найшвидше в Україні (і в світі) зараз розвивається ІТ-галузь. Потреба у відповідних фахівцях щороку зростає. Але, щоб знайти своє місце, треба знати загальноприйняті технології та інструментарій, засоби проектування та розробки програмних систем. Це ж стосується і розробок в біомедичній інженерії, оскільки зараз важко уявити біомедичні прилади і системи без якогось програмного забезпечення. Крім того дуже корисно використовувати ці технології і стандарти при дипломному проектуванні.

Предмет дисципліни

Навчальна дисципліна «Технологія створення програмних продуктів» вивчає комплекс взаємопов'язаних технологій, заснованих на процесному підході до інженерії вимог,

проектування, конструювання, тестування та супроводу складних програмних систем, згідно відповідно обраних моделей життєвого циклу.

Підготовка студентів проводиться на основі використання примірників учбової літератури технічної бібліотеки НТУУ «КПІ» та факультету, методичних матеріалів курсу на базі Платформи дистанційного навчання "Сікорський", додаткових учбових посібників, що надаються їм в електронному вигляді, а саме:

Кисельова О.Г., Соломін А.В. Навчальний посібник. Технологія створення програмних продуктів. Програмування в LabVIEW. - Київ, 2012 р.

Кисельова О.Г., Соломін А.В. Навчальний посібник. Технологія розробки віртуальних приладів. Програмування в LabVIEW. - Київ, 2014 р.

Вивчення теоретичного матеріалу лекцій супроводжується розглядом прикладів з використанням можливостей пакету NI LabVIEW та інших при підготовці до практичних робіт. В процесі викладання дисципліни використовується пояснювально-ілюстративний, пояснювально-спонукальний та частково-пошуковий метод викладання.

Перевірка засвоєння матеріалу проводиться у ході проведення практичних занять, з використанням комп'ютерного тестування.

Під час навчання та для взаємодії зі студентами використовуються сучасні інформаційно-комунікаційні та мережеві технології, розроблений та постійно вдосконалюється відповідний он-лайн курс на платформі "Сікорський".

Програмні результати навчання:

Загальні компетентності (ОП введено в дію Наказом ректора НОН/75/2022 від 15.02.2022 р.):

ЗК 1 - Здатність застосовувати знання у практичних ситуаціях.

ЗК 2 - Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК 4 - Навички використання інформаційних і комунікаційних технологій.

ЗК 6 - Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК 7 - Здатність генерувати нові ідеї (креативність).

ЗК 8 - Здатність приймати обґрунтовані рішення.

Спеціальні (фахові) компетентності (ОП введено в дію Наказом ректора НОН/75/2022 від 15.02.2022 р.):

ФК 1 - Здатність застосовувати пакети інженерного програмного забезпечення для проведення досліджень, аналізу, обробки та представлення результатів, а також для автоматизованого проектування медичних приладів та систем.

ФК 3 - Здатність вивчати та застосовувати нові методи та інструменти аналізу, моделювання, проектування та оптимізації медичних приладів і систем.

ФК 7 - Здатність планувати, проектувати, розробляти, встановлювати, експлуатувати, підтримувати, технічно обслуговувати, контролювати і координувати ремонт приладів, обладнання та системи для профілактики, діагностики, лікування і реабілітації, що використовується в лікарнях і науково-дослідних інститутах.

ФК 10 - Здатність застосовувати принципи побудови сучасних автоматизованих систем управління виробництвом медичних приладів, їх технічне, алгоритмічне, інформаційне і програмне забезпечення.

Програмні результати навчання після вивчення дисципліни «Технологія створення програмних продуктів» (ОП введено в дію Наказом ректора НОН/75/2022 від 15.02.2022 р.):

Інтегральна компетентність (ІК) - Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у біомедичній інженерії або у процесі навчання, що передбачає застосування певних теорій та методів хімічної, біологічної та медичної інженерії, і характеризується комплексністю та невизначеністю умов.

ПРН 5 - Вміти використовувати бази даних, математичне і програмне забезпечення для обробки даних та комп'ютерного моделювання біотехнічних систем.

ПРН 13 - Вміти аналізувати сигнали, які передаються від органів на прилади, та проводити обробку діагностичної інформації (сигнали та зображення).

ПРН 14 - Вміти аналізувати рівень відповідності сучасним світовим стандартам, а також оцінювати рішення і складати завдання на розробку автоматизованих систем управління з урахуванням можливостей сучасних технічних і програмних засобів автоматизації медичного обладнання.

ПРН 20 - Знання та використання методів дослідження об'єктів біомедичної інженерії, методів і засобів систематизації та обробки експериментальної інформації, методів статистичної обробки для моделювання та симуляції процесів і систем фізичної та біологічної природи, сучасних технологій програмування та інструментарію, які підтримують їх використання, методів проектування цифрових та мікропроцесорних систем медичного призначення.

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

За структурно-логічною схемою програми підготовки фахівця дисципліна «Технологія створення програмних продуктів» тісно пов'язана з іншими дисциплінами блоку інформатики: «Основи інформатики», «Основи дискретної математики». Їй безпосередньо передують дисципліна «Об'єктно-орієнтоване програмування».

Необхідні знання основ інформатики, об'єктно-орієнтованого програмування, програмного забезпечення.

Отримані практичні навички та засвоєні теоретичні знання під час вивчення навчальної дисципліни «Технологія створення програмних продуктів» можна використовувати в подальшому при проходженні переддипломної практики, для підготовки дипломних робіт (проектів) за спеціальністю та в подальшій практичній роботі за фахом.

3. Зміст навчальної дисципліни

Основні розділи та теми, що розглядатимуться в процесі вивчення курсу:

1. Вступ. Історичний і соціальний контекст програмної інженерії. Дисципліни та зміст програмної інженерії. Загальні принципи розробки програмних продуктів.

- Історичні аспекти розвитку програмного забезпечення та технології створення програмних продуктів.
- Дисципліни та проблеми програмної інженерії.
- Загальні принципи розробки програмних систем.

2. Ядро знань SWEBOOK. Характеристика областей знань з інженерії програмного забезпечення.

- Ядро знань SWEBOOK – узагальнення досвіду і основа стандартів в галузі програмної інженерії.
- Структура SWEBOOK.
- Характеристика областей знань з інженерії програмного забезпечення.

3. Процеси життєвого циклу (ЖЦ) розробки програмного забезпечення (ПЗ). Водоспадна (каскадна, класична) модель життєвого циклу розробки ПЗ.

- Поняття життєвого циклу ПЗ.
- Процеси життєвого циклу.
- Моделі життєвого циклу.
- Основні поняття і процеси водоспадної моделі ЖЦ.
- Особливості водоспадної моделі. Приклади застосувань.
- Переваги та недоліки.

4. Інкрементна та еволюційна моделі життєвого циклу розробки ПЗ.

- Основні поняття і процеси інкрементної моделі ЖЦ.
- Особливості інкрементної моделі. Приклади застосувань.
- Основні поняття і процеси еволюційної моделі ЖЦ.
- Особливості еволюційної моделі. Приклади застосувань.

- Переваги та недоліки.
5. Спіральна модель життєвого циклу розробки ПЗ. Agile-модель життєвого циклу розробки ПЗ.
- Основні поняття і процеси спіральної моделі ЖЦ.
 - Особливості спіральної моделі. Приклади застосувань.
 - Основні поняття і процеси Agile-моделі ЖЦ.
 - Особливості Agile-моделі. Приклади застосувань.
 - Переваги та недоліки.
6. Міжнародні та національні стандарти розробки складних програмних продуктів.
- Корпоративні, галузеві, державні та міжнародні стандарти.
 - Ядро знань SWEBOK, PMBOK і стандарти життєвого циклу ПЗ.
 - Основні стандарти програмної інженерії. Стандарти ISO/IEC 12207, ISO/IEC 15504, SEI CMM.
7. Основи проектування програмних систем. Модульність, інформаційна закритість, зв'язність.
- Аналіз і синтез при проектування програмних систем..
 - Структурування системи.
 - Моделювання управління.
 - Декомпозиція підсистем на модулі..
 - Модульність.
 - Інформаційна закритість.
 - Зв'язність і типи зв'язності
8. Зчепленість модулів, складність та якість ієрархічної структури програмних систем.
- Зчепленість модулів.
 - Метрики складності програмних систем.
 - Характеристики ієрархічної структури програмних систем.
9. Методології розробки ПЗ (RUP, MSF, XP, DSDM, RAD).
- Інженерія індустриального виробництва програмних продуктів.
 - Середовища для автоматизованого виробництва програмних продуктів.
 - Особливості методологій розробки ПЗ: RUP, MSF, XP, DSDM, RAD.
10. Архітектура ПЗ, стандарти опису архітектур ПЗ. Патерни проектування ПЗ
- Поняття архітектури ПЗ.
 - Проектування архітектури ПЗ, як один з етапів життєвого циклу.
 - Стандартизований, об'єктно-орієнтований, компонентний методи проектування архітектури.
 - Стандарти опису архітектур ПЗ.
 - Компонентне програмування.
 - Типи компонентних структур.
 - Шаблон (патерн), як розширення поняття „компонент”.
 - Методологія компонентної розробки ПЗ.
11. Засоби автоматизації розробки програмних продуктів. Якість ПЗ, метрики якості, стандарти якості ПЗ
- Автоматизоване проектування інформаційних систем з використанням CASE-технологій.
 - Класифікація CASE-технологій.
 - Інструментальні засоби підтримки технологій та їх класи.
 - Принципи організації проектування з використанням CASE засобів.
 - Аналіз функціональних можливостей CASE-засобів різних класів.
 - Модель якості програмних систем.
 - Стандартні показники якості.

- Зовнішні і внутрішні метрики якості.
- Керування якістю програмних систем.

12. Аналіз вимог замовника до ПЗ.

- Класифікація вимог.
- Аналіз і збирання вимог.
- Інженерія вимог.
- Фіксація вимог.
- Трасування вимог.
- Розробка технічного завдання.

13. Верифікація, валідація та тестування. Стандарти тестування ПЗ. Супровід програмних продуктів. Експлуатаційна, операційна, рекламна документація на ПЗ.

- Тестування ПЗ, як один з етапів життєвого циклу.
- Принципи верифікації, валідації та тестування програмних систем.
- Процеси і засоби тестування програмних компонентів.
- Функціональне тестування.
- Класифікація помилок і методи їх пошуку.
- Організація і методи супроводу програмних засобів.
- Організація документування програмних засобів.
- Вимоги до документації складних програмних засобів.
- Документування процесів і результатів сертифікації програмних продуктів.

Комп'ютерний практикум

Основні завдання циклу комп'ютерних практикумів: формування вмінь оптимального і ефективного застосування сучасних інформаційних технологій для вирішення задач комп'ютеризації в біомедичній інженерії, а також знайомство та опанування навичок роботи з відповідними комп'ютерними пакетами прикладних програм.

Теми практичних робіт:

1. Програмне середовище NI LabVIEW.
2. Редагування та налаштування віртуальних приладів.
3. Ієрархічна структура віртуальних приладів. Віртуальні прилади і підприлади.
4. Управління виконанням програм за допомогою структур. Цикли.
5. Управління виконанням програм за допомогою структур. Структури варіанту.
6. Масиви.
7. Кластери.
8. Графіки осцилограм (WAVEFORM CHART).
9. Графіки (WAVEFORM GRAPH, XY GRAPH).
10. Графіки інтенсивності.
11. Строкові дані.
12. Запис та зчитування файлів.

4. Навчальні матеріали та ресурси

Базова література

1. Навчальний посібник з дисципліни «Технології розробки програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія»/укл. Дегтярьова Л.М., Гроза П.М., Сомов С.В. – Полтава: ПолтНТУ, 2017. – 218 с. – Режим доступу:

http://reposit.nupp.edu.ua/bitstream/PolNTU/4431/1/%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D0%B8%D0%BA_%D0%A2%D0%A0%D0%9F%D0%97_%D0%BF%D1%80%D0%BE%D0%B2%D0%B5%D1%80%D0%B5%D0%BD%D0%BE-converted.pdf

2. Якість програмного забезпечення та тестування: базовий курс. Навчальний посібник / За ред. Крепич С.Я., Співак І.Я. / для бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення». – Тернопіль: ФОП Паляниця В.А., 2020. – 478с. – Режим доступу:
<http://dspace.wunu.edu.ua/bitstream/316497/39773/1/%D0%9D%D0%B0%D0%B2%D1%87%D0%0%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9%20%D0%BF%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA%20%D0%B7%20%D1%8F%D0%BA%D0%BE%D1%81%D1%82%D1%96%20%D0%9F%D0%97%20%D1%82%D0%B0%20%D1%82%D0%B5%D1%81%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F%20%281%29.pdf>
3. Карпенко М. Ю. Технології створення програмних продуктів та інформаційних систем : навч. посібник / М. Ю. Карпенко, Н. О. Манакова, І. О. Гавриленко ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2017. – 93 с. – Режим доступу:
<http://eprints.kname.edu.ua/46989/1/2017%20%D0%BF%D0%B5%D1%87.%2024%D0%9D%20%D0%9F%D0%BE%D1%81%D0%BE%D0%B1%D0%B8%D0%B514.04.17.pdf>
4. Авраменко А.С., Авраменко В.С., Косенюк Г.В. Тестування програмного забезпечення. Навчальний посібник. – Черкаси: ЧНУ імені Богдана Хмельницького, 2017. – 284 с. – Режим доступу:
<http://eprints.cdu.edu.ua/1482/1/testyvan.pdf>
5. Програмування в NI LabVIEW. Технологія розробки віртуальних приладів : навч. посіб. / О.Г. Кисельова, А.В. Соломін. – К. : НТУУ «КПІ», 2013. – 273 с. – Надсилається викладачем на електронну пошту групи.
6. Лавріщева К.М. Програмна інженерія : Підручник.–К.– 2008.–319 с. – Режим доступу:
www.cyb.univ.kiev.ua/library/books/lavrishcheva-6.pdf
7. Технологія створення програмних продуктів. Програмування в NI LabVIEW : навч. посіб. / О.Г. Кисельова, А.В. Соломін. – К.: НТУУ «КПІ», 2012. – 200 с. – Надсилається викладачем на електронну пошту групи.

Додаткова література

1. Карпенко С.Н. Введение в программную инженерию: Учебно-методические материалы по программе повышения квалификации «Информационные технологии и компьютерное моделирование в прикладной математике». – Нижний Новгород – 2007. – 103 с. – Режим доступу: <http://www.unn.ru/pages/issues/aids/2007/16.pdf>
2. Милованов И.В. Основы разработки программного обеспечения вычислительных систем : учебное пособие / И.В. Милованов, В.И. Лоскутов. – Тамбов: Изд-во ГОУ ВПО ТГТУ, 2011. – 88 с. – Режим доступу: <http://window.edu.ru/resource/507/76507/files/milovanov-t.pdf>
3. Вольфсон Б. Бесплатная электронная книга по гибким методологиям разработки. Режим доступу :
<https://dl.dropboxusercontent.com/u/4606839/Гибкие%20методологии%20разработки.pdf>
4. CASE-технологии. Современные методы и средства проектирования информационных систем : Электронный учебник. – Режим доступу: <http://bookz.ru/authors/avtor-neizvesten-3/casetechs.html>
5. Липаев В.В. Программная инженерия. Методологические основы: Учебник. – М.: Теис, 2006. – 608 с. – Режим доступу: <http://window.edu.ru/resource/438/57438>
6. Скопин И. Менеджмент программных проектов. Электронный учебник. – Режим доступу: http://bib.convdocs.org/v6046/скопин_и.н._основы_менеджмента_-программных_проектов._курс_лекций
7. LabVIEW для всех / Джеффри Тревис, М.: ДМК Пресс, 2005. – 544 с. (та інші редакції). – Режим доступу:
http://chaos.sgu.ru/library/programms/progr/labVIEW/LabVIEW_trevis.pdf

Інформаційні ресурси

1. <https://do.ipu.kpi.ua/course/view.php?id=2282> - Платформа дистанційного навчання "Сікорський"
 2. <http://www.ni.com/labview/> - Сайт компанії National Instruments
 3. www.picad.com.ua/lesson.htm - навчальні матеріали по LabVIEW
- В переліку інформаційних ресурсів наведено джерела їх отримання.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

№ з/п	Тема	Програмні результати навчання	Основні завдання	
			Контрольний захід	Термін виконання
1.	Тема 1. Вступ. Історичний і соціальний контекст програмної інженерії. Дисципліни та зміст програмної інженерії. Загальні принципи розробки програмних продуктів	ПРН 5, 14, 20	Практична робота 1	1-ий тиждень
2.	Тема 2. Ядро знань SWEBOOK. Характеристика областей знань з інженерії програмного забезпечення	ПРН 5, 14, 20	Практична робота 2	1-ий тиждень
3.	Тема 3. Процеси життєвого циклу (ЖЦ) розробки програмного забезпечення (ПЗ) Водоспадна (каскадна, класична) модель життєвого циклу розробки ПЗ	ПРН 5, 14, 20	Практична робота 3	2-ий тиждень
4.	Тема 4. Інкрементна та еволюційна моделі життєвого циклу розробки ПЗ	ПРН 5, 14, 20	Практична робота 4	2-ий тиждень
5.	Тема 5. Спіральна модель життєвого циклу розробки ПЗ. Agile-модель життєвого циклу розробки ПЗ	ПРН 5, 14, 20	Практична робота 5	3-ий тиждень
6.	Тема 6. Міжнародні та національні стандарти розробки складних програмних продуктів	ПРН 5, 14, 20	Практична робота 6	3-ий тиждень
7.	Тема 7. Основи проектування програмних систем. Модульність, інформаційна закритість, зв'язність	ПРН 5, 14, 20	Практична робота 7	4-ий тиждень

8.	Тема 8. Зчепленість модулів, складність та якість ієрархічної структури програмних систем	ПРН 5, 13, 14, 20	Практична робота 8	4-ий тиждень
9.	Тема 9 Методології розробки ПЗ (RUP, MSF, XP, DSDM, RAD)	ПРН 5, 13, 14, 20	Практична робота 9	5-ий тиждень
10	Тема 10. Архітектура ПЗ, стандарти опису архітектур ПЗ. Патерни проектування ПЗ	ПРН 5, 13, 14, 20	Практична робота 10	5-ий тиждень
11	Тема 11. Засоби автоматизації розробки програмних продуктів. Якість ПЗ, метрики якості, стандарти якості ПЗ	ПРН 5, 14, 20	Практична робота 11	6-ий тиждень
12	Тема 12. Аналіз вимог замовника до ПЗ.	ПРН 5, 14, 20	Практична робота 12	6-ий тиждень
13	Тема 13. Верифікація, валідація та тестування. Стандарти тестування ПЗ. Супровід програмних продуктів. Експлуатаційна, операційна, рекламна документація на ПЗ	ПРН 5, 13, 14, 20	МКР	7-ий тиждень
14	Модульна контрольна робота	ПРН 5, 13, 14, 20	МКР	7-ий тиждень
15	Залік	ПРН 5, 13, 14, 20	залік	8-ий тиждень

Платформа дистанційного навчання:

Для більш ефективної комунікації з метою розуміння структури навчальної дисципліни «Технологія створення програмних продуктів» і засвоєння матеріалу використовується електронна пошта, телеграм-канал, платформа дистанційного навчання "Сікорський" на основі системи Moodle КПІ-Телеком та сервіс для проведення онлайн-нарад Zoом, за допомогою яких:

- підвищується оперативність спілкування зі студентами, забезпечується зручний зворотній зв'язок;
- спрощується розміщення, доступ та обмін навчальним матеріалом;
- оцінюються навчальні завдання студентів;
- аналізується активність студентів.

6. Самостійна робота студента/аспіранта

Заплановано наступні види самостійної роботи: підготовка до аудиторних занять, обробка результатів практичних робіт та оформлення звіту, підготовка до модульної контрольної роботи, виконання реферату. Всього на самостійну роботу заплановано 66 годин.

Одним з основних видів семестрового контролю під час опанування навчальної дисципліни «Технологія створення програмних продуктів» є підготовка реферату. Реферат виконується згідно з вимогами, у термін, зазначений викладачем. Реферат — це науково-технічний документ, який містить вичерпну систематизовану інформацію за вибраною темою, передбачає виклад матеріалу на основі спеціально підібраної літератури та самостійно проведеного дослідження.

Студент може писати реферат тільки на погоджену з викладачем тему. Загальні вимоги до реферату:

- чіткість та логічна послідовність викладення матеріалу;
- переконливість аргументації;
- стислість і точність формулювань, які виключають можливість неоднозначного тлумачення;
- конкретність викладення результатів дослідження;
- обґрунтованість рекомендацій та пропозицій.

У рефераті повинні бути відображеними:

- актуальність тематики та відповідність до сучасного стану науки, техніки і питань виробництва;
- обґрунтування вибраного напрямлення досліджень, методів розв'язку задачі та їх порівняльні оцінки;
- аналіз та узагальнення існуючих результатів;
- розробка загальної методики проведення досліджень;
- характер і зміст виконаних теоретичних досліджень та розрахунків, методи досліджень;
- обґрунтування необхідності проведення експериментальних досліджень, принцип дії розроблених програм, характеристики цих програм, оцінка похибок розрахунків, отримані експериментальні дані;
- оцінка повноти розв'язку поставленої задачі;
- оцінка достовірності отриманих результатів, їх порівняння з аналогічними результатами;
- наукова та практична цінність виконаної роботи.

Структура реферату: титульний аркуш; зміст; перелік умовних позначень, символів, одиниць скорочень і термінів (за необхідності); вступ; суть реферату (основна частина); висновки; список використаних джерел (перелік посилань); додатки (за необхідності).

Приблизна тематика реферату:

1. Розробка моделі програмно-апаратного комплексу для моніторингу серцевого ритму.
2. Розробка засобів експрес-діагностики в середовищі NI LabVIEW.
3. Розробка віртуального інструменту генерації тестових звукових сигналів для діагностики слуху людини.
4. Розробка системи вейвлет-аналізу сигналів серця.
5. Розробка автоматизованого робочого місця лікаря, електронної картотеки пацієнтів.
6. Розробка віртуального приладу для діагностування дальтонізму людини.
7. Віртуальний прилад для обробки та підвищення якості рентгенівських зображень.
8. 3D-реконструкція біомедичних об'єктів по даним комп'ютерної томографії.
9. Віртуальний прилад для попередньої обробки мікроскопічних зображень в біомедицині.
10. Віртуальні прилади для статистичної обробки інформації в біомедицині.
11. Розробка системи вимірювання та аналізу артеріального тиску.
12. Розробка системи вимірювання та аналізу легеневого тиску.
13. Розробка віртуального електрокардіографу.
14. Розробка віртуального приладу для вимірювання швидкості поширення пульсової хвилі.
15. Розробка системи моніторингу температури внутрішніх органів.
16. Розробка віртуального приладу для вимірювання шумів серця.
17. Розробка системи для прогнозування групи крові, резус-фактору та статі дитини за показниками батьків.
18. Розробка віртуального обладнання для оцінки стану серцево-судинної системи людини.
19. Розробка системи для дослідження кардіоінтервалограми людини.
20. Розробка віртуального приладу для визначення швидкості реакції людини.
21. Розробка віртуального приладу для діагностування психічного стану людини (наприклад, по асоціаціях, що виникають при розгляді малюнків).
22. Віртуальний прилад для аналізу електрокардіограм.
23. Віртуальний прилад для підвищення якості електрокардіограм.

24. Віртуальний прилад для напівавтоматизованої обробки ехокардіограм.
25. Віртуальний прилад для напівавтоматизованої обробки та аналізу електроенцефалограм.
26. Віртуальний прилад для вимірювання та моніторингу шуму в приміщенні.

Титульний аркуш реферату повинен мати такий зміст: назва університету; назва факультету; назва кафедри; назва спеціальності, назва освітньо-професійної програми, назва навчальної дисципліни; тема реферату; прізвище та ім'я студента, курс, номер академічної групи, рік. За титульним аркушем слідує детальний план (зміст) реферату, в якому треба виділити вступ, розділи основного змісту (основні теми, що будуть розглядатися), їх підрозділи (за потребою), висновок, список використаних джерел. У змісті праворуч позначаються номери сторінок початку кожного питання. Кожен розділ починається з нової сторінки. Загальний обсяг реферату складає приблизно 20 сторінок основного тексту. Обсяг реферату визначається вмінням студента стисло і водночас вичерпно пояснити та проаналізувати. **Обов'язкова вимога: чітке посилання на джерела інформації.** Всі цифри, факти, думки вчених, цитати, формули повинні мати посилання у вигляді *2, с. 54+ (перша цифра означає номер джерела у наведеному в кінці творчої роботи списку літератури, а друга цифра – номер сторінки у цьому джерелі). Бажано використовувати таблиці, схеми, графіки, діаграми тощо. Список використаних джерел (не менше 10 джерел) оформляється згідно з діючими правилами. Якщо інформація взята з мережі Інтернет, потрібно, як і для звичайної літератури, вказати автора, назву статті, а потім навести адресу сайту в Інтернет. Реферат оцінюється за критеріями: логічності плану; повноти й глибини розкриття теми; достовірності отриманих даних; відображення практичних матеріалів; правильності формулювання заключень отриманих результатів та висновків; оформлення; обґрунтування власної думки студента з цього питання у вигляді висновку.

Граничний термін подання реферату на перевірку: 8-й тиждень навчання. Реферат не перевіряється на плагіат, але повинен відповідати вимогам академічної доброчесності. У разі виявлення академічної недоброчесності, робота анулюється і не перевіряється.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Відвідування занять

Відвідування лекційних занять не є обов'язковими, але бажаними, оскільки саме через оволодіння лекційного матеріалу формуються системні компетенції, які потім закріплюються на практичних заняттях.

Система оцінювання орієнтована на отримання балів за активність студента, а також виконання завдань, які здатні розвинути практичні уміння та навички.

Пропущене практичне заняття можна виконати і захистити на протязі тижня без штрафних балів (можливий дистанційний варіант). Інакше застосовується штрафний бал «-1».

Пропущені контрольні заходи

Пропущені контрольні заходи (захист практичних робіт) обов'язково відпрацьовуються на наступних заняттях за умови виконання завдання, яке заплановано на поточному занятті, або на консультаціях.

Пропущений контрольний захід (МКР) можна виконати на додатковому (консультаційному) занятті, але тільки у випадку пропуску з поважних причин.

Реферат, який подається на перевірку з порушенням терміну виконання, оцінюється зі зменшенням кількості вагових балів

Порушення термінів виконання завдань та заохочувальні бали

Заохочувальні бали		Штрафні бали	
Критерій	Ваговий бал	Критерій	Ваговий бал
Виконання практичних робіт	5 (за кожную практичну роботу)	Порушення термінів виконання практичних робіт (за кожную таку роботу)	-1
Використання нових технологій, не передбачених в навчальній програмі, при виконанні реферату і практичних робіт	+1	Порушення термінів виконання реферату (не пізніше, ніж за 10 днів до закінчення семестру)	-1 бал за кожен день
Проходження додаткових дистанційних курсів за темами, які погоджено з викладачами	+5		
Оформлення наукової роботи для участі у конкурсі студентських наукових робіт	+10		
Написання тез, статті, участь у міжнародних, всеукраїнських та/або інших заходах або конкурсах за тематикою навчальної дисципліни	+5		

** якщо контрольний захід був пропущений з поважної причини (хвороба, яка підтверджена довідкою встановленого зразку) – штрафні бали не нараховуються.*

Академічна доброчесність

Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

Норми етичної поведінки

Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

Процедура оскарження результатів контрольних заходів

Студенти мають можливість підняти будь-яке питання, яке стосується процедури контрольних заходів та очікувати, що воно буде розглянуто згідно із наперед визначеними процедурами.

Студент має право оскаржити результати контрольного заходу згідно затвердженого положення Про апеляції в КПІ імені Ігоря Сікорського (затверджено наказом №НОН/128/2021 від 20.05.2021 р.) - <https://osvita.kpi.ua/index.php/node/182>

Інклюзивне навчання

Навчальна дисципліна «Технологія створення програмних продуктів» може викладатися для більшості студентів з особливими освітніми потребами, окрім студентів з серйозними вадами

зору, які не дозволяють виконувати завдання за допомогою персональних комп'ютерів, ноутбуків та/або інших технічних засобів.

Дистанційне навчання

Дистанційне навчання відбувається через Платформу дистанційного навчання «Сікорський». Дистанційне навчання через проходження додаткових он-лайн курсів за певною тематикою допускається за умови погодження зі студентами. У разі, якщо невелика кількість студентів має бажання пройти он-лайн курс за певною тематикою, вивчення матеріалу за допомогою таких курсів допускається, але студенти повинні виконати всі завдання, які передбачені у навчальній дисципліні.

Список курсів пропонується викладачем після виявлення бажання студентами (оскільки банк доступних курсів поновлюється майже щомісяця).

Студент надає документ, що підтверджує проходження дистанційного курсу (у разі проходження повного курсу) або надає виконані практичні завдання з дистанційного курсу та за умови проходження усної співбесіди з викладачем за пройденими темами може отримати оцінки за контрольні заходи, які передбачені за вивченими темами. Виконання практичних робіт, а також виконання реферату, здійснюється під час самостійної роботи студентів у дистанційному режимі (з можливістю консультування з викладачем через електронну пошту, соціальні мережі).

Навчання іноземною мовою

Навчання англійською мовою здійснюється лише для студентів-іноземців.

За бажанням студентів, допускається вивчення матеріалу за допомогою англійських онлайн-курсів за тематикою, яка відповідає тематиці конкретних занять.

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

Види контролю та бали за кожен елемент контролю:

№ з/п	Контрольний захід	%	Ваговий бал	Кіл-ть	Всього
1.	Практична робота	60	5	12	60
2.	Реферат	20	20	1	20
3.	Модульна контрольна робота	20	20	1	20
	Всього				100

Результати оголошуються кожному студенту окремо у присутності або в дистанційній формі (на платформі Сікорський, е-кампус або е-поштою)

Критерії оцінювання на контрольних заходах:

№ з/п	Реферат	%	Ваговий бал	Кіл-ть	Всього
1.	Актуальність теми реферату	25	5	1	5
2.	Повнота висвітлення матеріалу	25	5	1	5
3.	Якість структури реферату, відповідність прийнятим нормам для науково-технічних матеріалів	25	5	1	5
4.	Якість оформлення реферату	25	5	1	5
	Всього				20

№ з/п	Модульна контрольна робота	%	Ваговий бал	Кіл-ть	Всього
1.	Повна і вичерпна відповідь (не менше 90% потрібної інформації)	100	5	4	20
2.	Несуттєві помилки у відповіді (не менше 75% потрібної інформації)	75	4	4	
3.	Є недоліки у відповіді та певні помилки (не менше 60% потрібної інформації)	60	3	4	
4.	Відповідь відсутня або не правильна	0	0	4	
	Максимальна кількість балів				20

У разі виявлення академічної недоброчесності під час дистанційного навчання – контрольний захід не враховується, студент до захисту не допускається.

Календарний контроль (КК) - Для студентів 4-го курсу не проводиться.

Семестрова атестація студентів

Обов'язкова умова допуску до заліку		Критерій
1	Поточний рейтинг	RD ≥ 60
2	Виконання практичних робіт	Захист практичних робіт
3	Виконання реферату	Захист реферату
4	Виконання МКР	Рейтинг МКР ≥ 10

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 для підготовки до модульної контрольної роботи, а також для підготовки до заліку наведено у додатку 1.

Дистанційне навчання через проходження додаткових он-лайн курсів за певною тематикою допускається за умови погодження зі студентами. У разі, якщо невелика кількість студентів має бажання пройти он-лайн курс за певною тематикою, вивчення матеріалу за допомогою таких курсів допускається, але студенти повинні виконати всі завдання, які передбачені у навчальній дисципліні. Список курсів пропонується викладачем після виявлення бажання студентами (оскільки банк доступних курсів поновлюється майже щомісяця). Студент надає документ, що підтверджує проходження дистанційного курсу (у разі проходження повного курсу) або надає виконані практичні завдання з дистанційного курсу та за умови проходження усної співбесіди з

викладачем за пройденими темами може отримати оцінки за контрольні заходи, які передбачені за вивченими темами.

Робочу програму навчальної дисципліни (силабус):

Складено: доцент, канд. фіз.-мат. наук, Соломін Андрій Вячеславович

Ухвалено кафедрою біомедичної інженерії (протокол № 1 від 29.08.2022 року);

Погоджено Методичною комісією факультету (протокол № 1 від 01.09.2022 року)

**Перелік запитань для підготовки до модульної контрольної роботи,
а також для підготовки до заліку**

Питання I

1. Навести зміст 3-х криз в галузі програмної інженерії та проаналізувати їх наслідки для розвитку технологій створення програмних продуктів
2. Перелічити відомі Вам моделі життєвого циклу програмного продукту та їх головні особливості
3. Охарактеризувати коротко класичну (каскадну, водоспадну) модель життєвого циклу програмного продукту, її переваги та недоліки в порівнянні з іншими моделями
4. Охарактеризувати коротко інкрементну модель життєвого циклу програмного продукту, її переваги та недоліки в порівнянні з іншими моделями
5. Охарактеризувати коротко V-образну модель життєвого циклу програмного продукту, її переваги та недоліки в порівнянні з іншими моделями
6. Охарактеризувати коротко еволюційну модель життєвого циклу програмного продукту, її переваги та недоліки в порівнянні з іншими моделями
7. Охарактеризувати коротко спіральну модель життєвого циклу програмного продукту, її переваги та недоліки в порівнянні з іншими моделями
8. Охарактеризувати коротко XP (екстремальну) модель життєвого циклу програмного продукту, її переваги та недоліки в порівнянні з іншими моделями
9. Які переваги в технології створення програмних продуктів з'явилися внаслідок використання компонентів повторного використання, об'єктно-орієнтованого програмування та поняття життєвого циклу програмного продукту?
10. Коротко охарактеризувати дисципліни програмної інженерії
11. Визначити сутність та призначення SWEBOOK
12. Коротко визначити зміст та призначення PMBOOK
13. Проаналізувати основні галузі знань програмної інженерії згідно SWEBOOK
14. Проаналізувати організаційні галузі знань програмної інженерії згідно SWEBOOK
15. Навести основні стандарти програмної інженерії та описати їх зв'язок з ядром знань SWEBOOK
16. Пояснити, що таке життєвий цикл програмного продукту, і описати основні процеси життєвого циклу

Питання II

1. Описати основні розділи технічного завдання. Яке основне призначення технічного завдання? Хто і з якою метою його складає?
2. Що таке модуль? Що таке модульність системи? Яким умовам повинен задовольняти ідеальний модуль і чому?
3. Розкрити принцип інформаційної закритості при проектуванні програмних систем
4. Що таке зв'язність модуля? Для яких цілей вона обчислюється і до яких її значень (якнайбільших чи якнайменших) треба прагнути і чому?
5. Що таке зчеплення модуля? Для яких цілей воно обчислюється і до яких його значень треба прагнути?
6. Як оцінюється складність програмної системи?
7. Проаналізувати і порівняти методи „висхідної розробки” і „низхідної розробки” програмної системи, їх переваги і недоліки
8. Проаналізувати і порівняти конструктивний і архітектурний методи розробки програмної системи, їх переваги і недоліки

9. Які існують типи зчеплення модулів в програмних системах? Їх міри (тобто яким параметром вимірюються) і до чого треба прагнути при проектуванні?
10. Пояснити, чому зв'язність модулів в програмних системах має бути максимально можливою, а зчеплення - мінімальним
11. Розкрити поняття «Нев'язка системи»
12. Визначити і порівняти «функціональні вимоги» і «нефункціональні вимоги» в технічних завданнях до програмних систем. Навести приклади
13. Описати, що розуміється під терміном „супровід програмного продукту”
14. Охарактеризувати діаграму SADT, її призначення, основні елементи. Навести приклади
15. Охарактеризувати діаграму ERD, її призначення, основні елементи. Навести приклади
16. Охарактеризувати діаграму DFD, її призначення, основні елементи. Навести приклади
17. Охарактеризувати діаграму USE CASE, її призначення, основні елементи. Навести приклади
18. Проаналізувати основні відмінності структурного і об'єктно-орієнтованого методів програмування
19. Порівняти наступні методи тестування програмного забезпечення: «метод чорного ящика», «метод білого ящика», «метод сірого ящика»
20. Дати визначення поняттю „CASE-засоби” в технології створення програмних продуктів. З якою метою вони застосовуються? Навести приклади

Питання III

1. Проаналізувати відомі Вам методи і інструменти для налаштування (пошуку несправностей) при розробці віртуальних приладів в LabVIEW
2. Пояснити відмінності застосування графіків Waveform graph і Waveform chart в LabVIEW
3. Проаналізувати і порівняти типи даних, які повинні подаватись на графіку Waveform chart, Waveform graph і XY-Graph в LabVIEW
4. Описати, що відбувається з розмірністю масивів на вхідному і вихідному індексуєчому тунелі циклів в LabVIEW (збільшується, зменшується або не міняється).
5. Описати, що таке масив. Якого типу дані (чисельні, кластери і так далі) можуть бути в масивах? А якого не можуть?
6. Для чого в LabVIEW використовуються структури (Case, While Loop, For Loop)? Намалюйте блок-схеми роботи кожної з цих структур
7. Чи може цикл по умові (While Loop) в LabVIEW не виконатися жодного разу? А цикл з фіксованим числом ітерацій (For Loop)? Поясніть
8. Пояснити, як Ви розумієте принцип управління потоком даних в LabVIEW
9. Пояснити, що таке зсувний регістр в структурі циклу LabVIEW, для чого він застосовується і навіщо його слід ініціалізувати
10. Обґрунтувати вимоги, які пред'являються до віртуального приладу LabVIEW, аби його можна було використовувати як віртуальний підприлад
11. Описати принцип дії системи обробки помилок у віртуальних приладах LabVIEW. Що таке кластер помилок?
12. Пояснити, з якою метою застосовуються дані типу Time Stamp в LabVIEW
13. Порівняти індексуєчий і неіндексуєчий тунелі в структурі циклу в LabVIEW
14. Описати, що таке кластер в LabVIEW. З яких елементів він може збиратися?
15. Пояснити, що таке - властивість поліморфізму функцій в LabVIEW

Питання IV

Тестова задача по роботі з блок-діаграмами NI LabVIEW