

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДВНЗ «ПРИКАРПАТСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАСИЛЯ СТЕФАНІКА»**

Фізико-технічний факультет
Кафедра комп'ютерної інженерії та електроніки

**СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СПЕЦІАЛІЗОВАНИХ КОМП'ЮТЕРНИХ
СИСТЕМ**

Освітня програма Комп'ютерна інженерія
Галузь знань 12 Інформаційні технології
Спеціальність 123 Комп'ютерна інженерія

Затверджено на засіданні кафедри
Протокол № 1 від “26” серпня 2020 р.

Івано-Франківськ – 2021 рік

ЗМІСТ

1. Загальна інформація
2. Анотація до курсу
3. Мета та цілі курсу
4. Компетентності
5. Результати навчання
6. Організація навчання курсу
7. Система оцінювання курсу
8. Політика курсу
9. Рекомендована література

1. Загальна інформація	
Назва дисципліни	Програмне забезпечення спеціалізованих комп'ютерних систем
Рівень вищої освіти	Другий рівень вищої освіти
Викладач (-і)	доцент, кандидат технічних наук Грига Володимир Михайлович
Контактний телефон викладача	0342596007
Е-mail викладача	volodymyr.gryga@pnu.edu.ua
Формат дисципліни	Семестровий
Обсяг дисципліни	3 кредити
Посилання на сайт дистанційного навчання	http://www.d-learn.pnu.edu.ua/
Консультації	відповідно до графіку індивідуальних консультацій, який розміщений на інформаційному стенді кафедри комп'ютерної інженерії та електроніки
2. Анотація до курсу	
<p>Дисципліна «Програмне забезпечення спеціалізованих комп'ютерних систем» належить до переліку обов'язкових освітніх компонент за освітнім рівнем «магістр», що пропонуються в рамках циклу професійної та практичної підготовки студентів за освітньо-професійною програмою «Комп'ютерна інженерія». Вона забезпечує формування у студентів науково-дослідницьких і професійно-орієнтованих компетенцій. Предметом вивчення навчальної дисципліни є засвоєння основних етапів розробки та тестування програмного забезпечення спеціалізованих комп'ютерних систем з використанням високорівневих мов програмування, зокрема SystemC.</p> <p>Силабус навчальної дисципліни «Програмне забезпечення спеціалізованих комп'ютерних систем» складений відповідно до освітньо-професійної програми «Комп'ютерна інженерія» підготовки магістрів спеціальності 123 «Комп'ютерна інженерія».</p>	
3. Мета та цілі курсу	
<p>Мета: вивчення студентами основних етапів розробки програмного забезпечення СКС, видів та рівнів тестування ПЗ, опанування основних способів і методологій розробки програмного забезпечення СКС з використанням автоматизованих програмних систем та комплексів.</p> <p>У результаті вивчення навчальної дисципліни магістр повинен</p> <p>знати:</p> <ul style="list-style-type: none"> - основні поняття програмного забезпечення СКС; - класифікацію програмного забезпечення комп'ютерних систем; - етапи та методи розробки програмного забезпечення комп'ютерних систем; - класифікацію видів і методів тестування; - основні методи подання алгоритмів; - основні принципи ТЛМ методології для розробки і налаштування програмної та апаратної частини проекту; - методологію компіляції високорівневого опису моделі пристрою в код опису його архітектури; - етапи автоматизованого синтезу SystemC моделей; - основні принципи генерації триадресного коду. <p>вміти:</p> <ul style="list-style-type: none"> - розробляти програмне забезпечення для конкретних спеціалізованих систем та їх компонентів; - використовувати інструментальні засоби для розробки програмних моделей компонентів спеціалізованих комп'ютерних систем; 	

- тестувати програмне забезпечення компонентів СКС;
- розробляти тестувальні стенди та системи тестування програмних моделей;
- описувати роботу компонентів СКС на мові SystemC;
- визначати програмну складність алгоритмічних моделей систем;
- аналізувати структуру алгоритму системи;
- генерувати програмні моделі компонентів СКС вбудованими засобами розробки.

4. Компетентності

Спеціальні (фахові) компетентності

К9. Здатність до проектування і програмування. Здатність проектувати інформаційні та спеціалізовані комп'ютерні системи, системи на кристалі, цифрові системи, топології інтегральних схем для КМОН і БІ-МОН технологічних процесів, МЕМС, сенсори і актуатори, багатошарові друковані плати з використанням сучасних САПР, захищати інформацію, опрацьовувати аналогові і цифрові сигнали, працювати з програмним забезпеченням спеціалізованих комп'ютерних систем, програмувати мікроконтролери і пристрої зв'язку з об'єктом.

5. Результати навчання

Р94. Працювати з програмним забезпеченням комп'ютерних систем, знати архітектуру і програмування 8/16/32- розрядних мікроконтролерів, програмувати пристрої зв'язку з об'єктом на мові C/C++ та асемблера.

6. Організація навчання курсу

Обсяг курсу

Вид заняття	Загальна кількість годин
лекції	16
семінарські заняття / практичні / лабораторні	16
самостійна робота	58

Ознаки курсу

Семестр	Спеціальність	Курс (рік навчання)	Нормативний / вибірковий
III	123 Комп'ютерна інженерія	2	нормативна

Тематика курсу

Тема, план	Форма заняття	Літера- тура	Кіль- кість годин	Вага оцінки	Термін виконання
------------	------------------	-----------------	-------------------------	----------------	------------------

Змістовий модуль 1. Методи розробки та тестування програмного забезпечення СКС

Тема 1. Вступ. Основні поняття програмного забезпечення СКС. Мови програмування компонентів комп'ютерних систем та їх розвиток.	лекція	[1-15]	2	2	Згідно розкладу
Тема 2. Етапи розробки ПЗ. Методи розробки програмного забезпечення. Ітеративна та інкрементна розробка.	лекція	[1-15]	2	2	Згідно розкладу
Тема 3. Тестування програмного забезпечення. План тестування. Рівні та види тестування. Тестувальні стенди. Системи	лекція	[1-15]	2	2	Згідно розкладу

тестування програмних моделей.					
Модульний контроль 1			8	7	Згідно розкладу
Змістовий модуль 2. Методологія розробки програмного забезпечення СКС з використанням засобів SystemC					
Тема 4. Моделі і системи. Моделювання програмних систем на рівні транзакції. Складність проектування. Програмна складність комп'ютерних систем.	лекція	[1-15]	2	2	Згідно розкладу
Тема 5. Абстрактні моделі. TLM методологія розробки і налаштування програмної та апаратної частини проекту. Моделювання проектів засобами SystemC. Модулі та процеси.	лекція	[1-15]	2	2	Згідно розкладу
Тема 6. Синтез SystemC моделей в Agility. Підмножина C++ для синтезу моделей. Приклади проектів в SystemC.	лекція	[1-15]	2	2	Згідно розкладу
Тема 7. Система введення алгоритму. Визначення та аналіз структури алгоритму. Програмний комплекс ОСА.	лекція	[1-15]	2	1	Згідно розкладу
Тема 8. Автоматизовані системи проектування алгоритмічних операційних пристроїв з графічного подання виконуваних алгоритмів. Програмна система ОСКАР.	лекція	[1-15]	2	1	Згідно розкладу
Модульний контроль 2			8	6	Згідно розкладу
Лабораторні роботи					
Тема 1. Моделювання інтерфейсу в SystemC на основі семафору.	Лаб. робота	[1-15]	2	4	Згідно розкладу
Тема 2. Реалізація та моделювання оперативної пам'яті в SystemC.	Лаб. робота	[1-15]	2	4	Згідно розкладу
Тема 3. Реалізація та моделювання буфера FIFO в SystemC.	Лаб. робота	[1-15]	2	4	Згідно розкладу
Тема 4. Реалізація та моделювання асинхронної пам'яті в SystemC.	Лаб. робота	[1-15]	2	4	Згідно розкладу
Тема 5. Реалізація та моделювання керуючого автомату Мура в SystemC.	Лаб. робота	[1-15]	4	4	Згідно розкладу

Тема 6. Реалізація проекту в середовищі розробки програмовних систем на кристалі PSoC Designer.	Лаб. робота	[1-15]	4	5	Згідно розкладу
Контроль лабораторних робіт			16	25	
Самостійна робота студентів					
Тема 1. Класифікація програмного забезпечення КС. Системне, прикладне та інструментальне ПЗ.	Самостійна робота	[1-15]	8	2	Впродовж семестру
Тема 2. Гнучкі методи розробки ПЗ. Каскадна і спіральна моделі ПЗ. Класи програмного забезпечення.	Самостійна робота	[1-15]	8	2	Впродовж семестру
Тема 3. Тестувальні стандарти. Ручне та автоматизоване тестування. Написання тест-кейсів. Статичне та динамічне тестування.	Самостійна робота	[1-15]	8	1	Впродовж семестру
Контроль самостійної роботи			64	6	Згідно розкладу
Тема 4. Паралельні процеси в мові SystemC. Компілювання SystemC моделі. Генератори тестів.	Самостійна робота	[1-15]	8	2	Впродовж семестру
Тема 5. Моделювання систем мовою C++. Компілятори синтезу конструкцій мови SystemC. Підмножина OSCI.	Самостійна робота	[1-15]	8	2	Впродовж семестру
Тема 6. Симуляційні ядра. Ієрархічні канали. Метод нотифікації та примітивні канали в SystemC.	Самостійна робота	[1-15]	8	2	Впродовж семестру
Тема 7. Генератори VHDL-моделей компонентів СКС. Опис алгоритму мовами Mitrion-C та Impulse-C.	Самостійна робота	[1-15]	8	1	Впродовж семестру
Тема 8. Компілятор Map Compiler. Оптимізація та розпаралелювання програмного коду. Режими компіляції. Динамічна бібліотека.	Самостійна робота	[1-15]	8	1	Впродовж семестру
Контроль самостійної роботи			56	6	Згідно розкладу
Підсумковий контроль (екзамен)				50	
7. Система оцінювання курсу					
Загальна система оцінювання курсу	Поточний контроль здійснюється під час проведення лекційних, лабораторних, індивідуальних занять і має на меті				

перевірку знань студентів з окремих тем навчальної дисципліни та рівня їх підготовленості до виконання конкретної роботи. Оцінки у національній шкалі («відмінно» - 5, «добре» - 4, «задовільно» - 3, «незадовільно» - 2), отримані студентами, виставляються у журналах обліку відвідування та успішності академічної групи.

Модульний контроль (сума балів за окремий змістовий модуль) проводиться (виставляється) на підставі оцінювання результатів знань студентів після вивчення матеріалу з логічно завершеної частини дисципліни – змістового модуля.

Завданням модульного контролю є перевірка розуміння та засвоєння певного матеріалу (теми), вироблення навичок проведення розрахункових робіт, вміння вирішувати конкретні ситуативні задачі, самостійно опрацьовувати тексти, здатності осмислювати зміст даної частини дисципліни, уміння публічно чи письмово подати певний матеріал.

Семестровий (підсумковий) контроль проводиться у формі екзамену.

Екзамен – форма підсумкового контролю, яка передбачає перевірку розуміння студентом теоретичного та практичного програмного матеріалу з усієї дисципліни, здатності творчо використовувати здобуті знання та вміння, формувати власне ставлення до певної проблеми тощо.

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
80 – 89	B	добре	
70 – 79	C		
60 – 69	D	задовільно	
50 – 59	E		
26 – 49	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-25	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

Вимоги до письмової роботи

Підсумкова письмова робота виконується у формі тестових завдань з вибором правильної відповіді. Кількість тестових завдань – 25.

Лабораторні заняття

Після узагальнення (вступного слова) викладач дає відповіді на окремі теоретичні запитання, які виникли в студентів у процесі підготовки до заняття. Зазвичай з кожної теми лекційного курсу на практичні заняття виносять індивідуалізовані теми комплексного характеру, які дають змогу студенту ширше застосувати здобуті знання та підготуватися до самостійного виконання домашнього завдання.

Для перевірки рівня засвоєння навчального матеріалу

	<p>студенти виконують тестові завдання.</p> <p>До початку лабораторної роботи студент має отримати допуск на основі усної співбесіди. На лабораторній роботі кожен студент отримує інструкцію до виконання. Після завершення роботи студент оформляє і захищає звіт з результатами роботи.</p>
<p>Умови допуску до підсумкового контролю</p>	<p>Студент допускається до складання екзамену, якщо впродовж семестру він за змістові модулі набрав сумарно 25 балів і вище.</p> <p>Студент не допускається до складання екзамену, якщо впродовж семестру він за змістові модулі набрав менше 25 балів. У цьому випадку студенту у відомості робиться запис "не допущений" і виставляється набрана кількість балів. Допускається, як виняток, з дозволу декана факультету за заявою, погодженою з відповідною кафедрою, одноразове виконання студентом додаткових видів робіт з навчальної дисципліни (відпрацювання пропущених занять, перескладання змістових модулів, виконання індивідуальних завдань тощо) для підвищення оцінок за змістові модулі.</p> <p>Напередодні екзамену викладач подає доповідну декану про недопуск студентів академічної групи (груп). Відмітка про недопуск у відомості робиться при наявності розпорядження декана.</p>
<p>8. Політика курсу</p>	
<p>Студент зобов'язаний відвідувати заняття відповідно до встановленого розкладу, не запізнюватися, мати відповідний зовнішній вигляд. У разі відсутності через хворобу надається відповідна довідка.</p> <p>Пропущена лекція відпрацьовується студентом самостійно, як короткий конспект за темою заняття.</p> <p>Пропущена лабораторна робота виконується студентом самостійно вдома або в комп'ютерному класі, результати оцінюються викладачем.</p> <p>У випадку, коли студент приймав участь у програмі мобільності, можливе врахування отриманих оцінок в іншому навчальному закладі за умови відповідності навчальних планів.</p> <p>Можливе зарахування результатів неформальної освіти згідно з Положенням про порядок зарахування результатів неформальної освіти у ДВНЗ «Прикарпатський національний університет імені Василя Стефаника».</p> <p>Політика академічної поведінки і етики</p> <p>Студент повинен бути толерантним і поважати думку інших.</p> <p>Заперечення повинні формулюватися тільки в коректній формі.</p> <p>Плагіат та академічна недоброчесність несумісні з принципами діяльності ЗВО.</p> <p>Не допускається підказування та списування під час здачі будь-яких робіт поточного, рубіжного чи підсумкового контролю.</p> <p>Не допускається користування телефонами та будь-якими іншими електронними засобами під час здачі будь-яких робіт поточного, рубіжного, чи підсумкового контролю.</p>	
<p>9. Рекомендована література</p>	
<p style="text-align: center;">Базова</p> <ol style="list-style-type: none"> 1. Русанова О.В., Корочкін О.В. Програмне забезпечення комп'ютерних систем. Програмування та компіляція: Навч. посібник . [Електронний ресурс] / О.В.Корочкін, Русанова О.В. – Київ : КПІ ім. Ігоря Сікорського, 2020. – 95 с. 2. Білас О.Є. Якість програмного забезпечення та тестування. Навчальний посібник– Львів: Видавництво Львівської політехніки, 2011. – 216 с. 3. Ю.І. Грицюк Аналіз вимог до програмного забезпечення. Навчальний посібник– Львів: Видавництво Львівської політехніки, 2018. – 456 с. 4. Буров Є.В. Концептуальне моделювання інтелектуальних програмних систем. Монографія – Львів: Видавництво Львівської політехніки, 2012. – 432 с. 	

5. А.О. Мельник, В.А. Мельник Персональні суперкомп'ютери: архітектура, проектування, застосування: монографія. – Львів: Видавництво Львівської політехніки, 2013. – 516 с.
6. Лайза Криспин, Джанет Грегори. Гибкое тестирование: практическое руководство для тестировщиков ПО и гибких команд. — М.: «Вильямс», 2010. — 464 с.
7. Канер Кем, Фолк Джек, Нгуен Енг Кек. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений. — Киев: ДиаСофт, 2001. — 544 с.
8. Калбертсон Роберт, Браун Крис, Кобб Гэри. Быстрое тестирование. — М.: «Вильямс», 2002. — 374 с.
9. Синицын С. В., Налютин Н. Ю. Верификация программного обеспечения. — М.: БИНОМ, 2008. — 368 с.
10. V.Kumar, A.Grama, A.Gupta, G.Karypis. Introduction to Parallel Computing. Design and Analysis of Algorithms.- Benjamin/Cummings Pub.Co, 2013.-597 p.

Допоміжна

11. Корнеев В.В. Параллельные вычислительные системы.-М:Мир, 2009.
12. Бейзер Б. Тестирование чёрного ящика. Технологии функционального тестирования программного обеспечения и систем. — СПб.: Питер, 2004. — 320 с.
13. Томас Бройнль. Паралельне програмування: Початковий курс.-К.:Вища школа, 1997.- 358 с.
14. Вальковский В.А. Распараллеливание алгоритмов и программ. Структурный подход.- М.: Радио и связь, 1989.-176 с.
15. Watson P., Robinson E.H. The Hardware Architecture of the ICL Goldrush MegaServer.Ungenuity. The ICL Technical Journal,1995.10(2): p.206-219

Викладач



Грига В.М.