

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МИКОЛАЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ В. О. СУХОМЛИНСЬКОГО
Механіко-математичний факультет

Кафедра комп'ютерних наук та прикладної математики




РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ПРОГРАМУВАННЯ

Ступінь бакалавра

Галузь знань 11 Математика та статистика
спеціальність 113 «Прикладна математика»
освітня програма Інформатика

2019 – 2020 навчальний рік

Розробник: Булгакова Олександра Сергіївна, доцент кафедри комп'ютерних наук та прикладної математики, кандидат технічних наук, доцент

 (Булгакова О.С)

Робоча програма затверджена на засіданні кафедри комп'ютерних наук та прикладної математики

Протокол № 1 від «27» серпня 2019 р.

Завідувач кафедри  (Поздєєв В.О.)

«27» серпня 2019 р.

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, освітній ступінь	Характеристика навчальної дисципліни	
		<i>денна форма навчання</i>	
Кількість кредитів – 5	Галузь знань 11 Математика та статистика	Нормативна	
	Спеціальність 113 Прикладна математика		
Індивідуальне науково-дослідне завдання – (підготовка мультипрезентації одного з напрямів сучасних технологій програмування)	Освітня програма: Інформатика	<i>Рік підготовки:</i>	
		2-й	
Загальна кількість годин 150		<i>Семестр</i>	
		1-й	
Тижневих годин для денної форми навчання: аудиторних – 4 самостійної роботи студента - 8	Ступінь бакалавра	<i>Лекції</i>	
		20 год.	
		<i>Практичні, семінарські</i>	
		<i>Лабораторні</i>	
		30 год.	
		<i>Самостійна робота</i>	
100 год.			
		Вид контролю: іспит	

Мова навчання – українська.

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної та індивідуальної роботи становить: для денної форми навчання – 50 год. – аудиторні заняття, 100 год. – самостійна робота (30% ~ 70%).

2. Мета, завдання навчальної дисципліни та результати навчання

Мета курсу – ознайомлення студентів з сучасною мовою програмування C# та оволодіння основними можливостями цієї мови, навичками хорошого стилю програмування, методами проектування та створення програм згідно сучасних технологій програмування.

Передумови для вивчення дисципліни: «Алгоритми та структури даних», «ООП».

Студент повинен знати:

основні елементи мови C# – типи даних, оператори та керуючі інструкції мови; основні принципи ООП, правила визначення класів в мові C#.

Студент повинен вміти:

побудувати алгоритм розв'язку задачі відповідної складності, порівняти різні алгоритми і обрати оптимальний для конкретної задачі, створити консольне прикладення, що реалізує обраний алгоритм; налагодити та протестувати програму; використовуючи панель інструментів середовища, створити просте Windows-прикладення.

Очікувані результати навчання:

Демонструвати знання й розуміння основних концепцій, принципів, теорій прикладної математики і використовувати їх на практиці.

Поєднувати методи математичного та комп'ютерного моделювання з неформальними процедурами експертного аналізу для пошуку оптимальних рішень.

Будувати ефективні щодо точності обчислень, стійкості, швидкодії та витрат системних ресурсів алгоритми для чисельного дослідження математичних моделей та розв'язання практичних задач

Згідно з вимогами ОПП студент оволодіває такими *компетентностями*:

I. Загальнопредметні:

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК06. Здатність до абстрактного мислення, аналізу та синтезу.

II. Фахові:

ФК07. Здатність експлуатувати та обслуговувати програмне забезпечення автоматизованих та інформаційних систем різного призначення.

ФК08. Здатність використовувати сучасні технології програмування та тестування програмного забезпечення.

3. Програма навчальної дисципліни

Кредит 1. Базові поняття програмування.

Тема 1. Основні поняття програмного забезпечення. Огляд процесу розробки програмного забезпечення. Етапи виконання комп'ютерної програми. Блок-схема та псевдокод як засоби, що використовуються у процесі розробки програми. Мови програмування. Процедурні та об'єктно-орієнтовані мови програмування. Місце мови C# серед сучасних мов програмування.

Тема 2. Представлення даних та двійкова арифметика. Необхідність типізації даних. Фундаментальні поняття типів даних та змінних. Змінні та константи. Правила іменування змінних. Операції. Використання програмного синтаксису.

Основні типи конструкцій в програмуванні. Використання розгалужень та циклів. Способи визначення логічних помилок. Культура програмування, поняття про ефективність алгоритму.

Тема 3. Процедури та функції як засіб створення модульних програм. Поняття аргументу, поняття параметра. Формальні і фактичні параметри, їх зв'язок за значенням та за

адресою. Вхідні та вихідні параметри. Повернення скалярного значення функцією. Рекурсія.

Кредит 2. Загальний огляд платформи Microsoft .NET

Тема 4. Огляд платформи Microsoft .NET. Огляд циклу розробки програмного забезпечення. Переваги платформи .NET.

Тема 5. Компоненти платформи .NET. Формат виконуваних програмних модулів (EXE і DLL), загальномовне виконуюча среда, Common Language Runtime, CLR, набір програмних засобів.

Кредит 3. Основи мови програмування C#

Тема 6. Структура програми мовою C#. Основні елементи програми. Основні операції вводу/виводу. Засоби форматування. Компілювання, виконання та відлагодження.

Тема 7. Огляд загальної системи типів. Використання вбудованих типів-значень даних у мові C#. Створення типів даних користувача в мові C# (зліченні типи та структури). Перетворення типів (явне та неявне). Використання пробілів та дужок у виразах для покращення читання програми.

Тема 8. Використання основних програмних конструкцій у мові C#. Розгалуження. Цикли. Поняття виключної ситуації. Обробка основних виключних ситуацій. Генерування виключних ситуацій.

Кредит 4. Поняття про класи та методи. Робота з масивами даних в мові C#

Тема 9. C# як об'єктно-орієнтована мова. Поняття про визначення класу та створення об'єкту. Типи-значення та типи-посилання. Визначення класу, доступ до даних-членів класу. Визначення методів класу.

Тема 10. Використання методів. Використання параметрів. Параметри за замовчуванням. Рекурсія.

Тема 11. Загальний огляд масивів як способу групування даних. Створення та використання масивів у мові C#. Використання конструкції new. Багатомірні масиви. Масиви як параметри методів. Методи, що повертають масиви. Цикл по елементах масива (foreach).

Кредит 5. Використання класів в мові C#

Тема 12. Створення об'єкту даного класу. Конструктор класу. Конструктор за замовчуванням та конструктор з параметрами. Створення параметризованих об'єктів класу.

Тема 13. Деструктор класу. Поняття про систему збору «сміття». Використання службового слова this.

Тема 14. Управління доступом до членів класу, інкапсуляція. Передача об'єктів методам класу. Правила використання модифікаторів ref- та out- для параметрів методів.

4. Структура навчальної дисципліни

Назви кредитів і тем	Кількість годин					
	усьо го	у тому числі				
		л	П	лаб	інд	ср
1	2	3	4	5	6	7
Кредит 1. Базові поняття програмування.						
Тема 1. Основні поняття програмного забезпечення.	8	2				6
Тема 2. Представлення даних та двійкова арифметика.	12	2				10

Тема 3. Процедури та функції як засіб створення модульних програм.	10			4		6
Усього	30					
Кредит 2. Загальний огляд платформи Microsoft .NET						
Тема 4. Огляд платформи Microsoft .NET.	15	2		4		9
Тема 5. Компоненти платформи .NET.	15	2		4		9
Усього	30					
Кредит 3. Основи мови програмування C#						
Тема 6. Структура програми мовою C#.	10	2		2		6
Тема 7. Огляд загальної системи типів.	10	2		2		6
Тема 8. Використання основних програмних конструкцій у мові C#.	10			2		8
Усього	30	4		6		20
Кредит 4. Поняття про класи та методи. Робота з масивами даних в мові C#						
Тема 9. Поняття про визначення класу та створення об'єкту.	10	2		2		6
Тема 10. Використання методів.	10	2		2		6
Тема 11. Загальний огляд масивів як способу групування даних.	10	2		4		4
Усього	30	6		8		16
Кредит 5. Використання класів в мові C#						
Тема 12. Створення об'єкту даного класу.	10	2		2		6
Тема 13. Деструктор класу.	10			2		8
Тема 14. Управління доступом до членів класу, інкапсуляція.	10					10
Усього	30	2		4		24
Усього годин:	150	20		30		100

5. Теми лекційних занять

N з/п	Назва теми	Кількість годин
<i>Кредит 1. Базові поняття програмування.</i>		
1	Тема 1. Основні поняття програмного забезпечення.	2
2	Тема 2. Представлення даних та двійкова арифметика.	2
<i>Кредит 2. Загальний огляд платформи Microsoft .NET</i>		
3	Тема 4. Огляд платформи Microsoft .NET.	2
4	Тема 5. Компоненти платформи .NET.	2
<i>Кредит 3. Основи мови програмування C#</i>		
5	Тема 6. Структура програми мовою C#.	2
6	Тема 7. Огляд загальної системи типів.	2
<i>Кредит 4. Поняття про класи та методи. Робота з масивами даних в мові C#</i>		
7	Тема 9. Поняття про визначення класу та створення об'єкту.	2
8	Тема 10. Використання методів.	2
9	Тема 11. Загальний огляд масивів як способу групування даних.	2
<i>Кредит 5. Використання класів в мові C#</i>		
10	Тема 12. Створення об'єкту даного класу.	2

	Разом	20
--	-------	----

6. Теми лабораторних занять

N з/п	Назва теми	Кількість годин
<i>Кредит 1. Базові поняття програмування.</i>		
1	Тема 3. Процедури та функції. Поняття аргументу, поняття параметра. Формальні і фактичні параметри, їх зв'язок за значенням та за адресою. Вхідні та вихідні параметри.	4
<i>Кредит 2. Загальний огляд платформи Microsoft .NET</i>		
2	Тема 4. Огляд платформи Microsoft .NET. Огляд циклу розробки програмного забезпечення.	4
3	Тема 5. Компоненти платформи .NET: EXE та DLL, CLR	4
<i>Кредит 3. Основи мови програмування C#</i>		
4	Тема 6. Структура програми мовою C#. Основні операції вводу/виводу. Засоби форматування.	2
5	Тема 7. Огляд загальної системи типів. Створення типів даних користувача в мові C#	2
6	Тема 8. Використання основних програмних конструкцій у мові C#. Розгалуження. Цикли.	2
<i>Кредит 4. Поняття про класи та методи. Робота з масивами даних в мові C#</i>		
7	Тема 9. Поняття про визначення класу та створення об'єкту. Типи-значення та типи-посилання. Визначення класу, доступ до даних-членів класу.	2
8	Тема 10. Використання методів, параметрів. Рекурсія.	2
9	Тема 11. Створення та використання масивів у мові C#.	4
<i>Кредит 5. Використання класів в мові C#</i>		
10	Тема 12. Конструктор класу за замовчуванням та конструктор з параметрами.	2
11	Тема 13. Деструктор класу. Використання службового слова this.	2
	Разом	30

7. Самостійна робота

N з/п	Назва теми	Кількість годин
<i>Кредит 1. Базові поняття програмування.</i>		
1	Тема 1. Процедурні та об'єктно-орієнтовані мови програмування. Місце мови C# серед сучасних мов програмування	6
2	Тема 2. Правила іменування змінних. Операції. Використання програмного синтаксису.	10
3	Тема 3. Формальні і фактичні параметри, їх зв'язок за значенням та за адресою. Вхідні та вихідні параметри. Повернення скалярного значення функцією.	6
<i>Кредит 2. Загальний огляд платформи Microsoft .NET</i>		
4	Тема 4. Набуття практичних навичок роботи в середовищі Visual Studio .NET. Переваги платформи .NET.	9

5	Тема 5. Компоненти платформи .NET. набір програмних засобів.	9
<i>Кредит 3. Основи мови програмування C#</i>		
6	Тема 6. Структура програми мовою C#. Засоби форматування. Компілювання, виконання та відлагодження.	6
7	Тема 7. Огляд загальної системи типів. Перетворення типів (явне та неявне).	6
8	Тема 8. Використання основних програмних конструкцій у мові C#. Поняття виключної ситуації. Обробка основних виключних ситуацій. Генерування виключних ситуацій.	8
<i>Кредит 4. Поняття про класи та методи. Робота з масивами даних в мові C#</i>		
9	Тема 9. Визначення класу, доступ до даних-членів класу. Визначення методів класу.	6
10	Тема 10. Параметри за замовчуванням. Рекурсія.	4
11	Тема 11. Загальний огляд масивів як способу групування даних. Багатовимірні масиви. Масиви як параметри методів. Методи, що повертають масиви. Цикл по елементах масива (foreach).	16
<i>Кредит 5. Використання класів в мові C#</i>		
12	Тема 12. Створення параметризованих об'єктів класу.	6
13	Тема 13. Деструктор класу. Поняття про систему збору «сміття».	8
14	Тема 14. Управління доступом до членів класу, інкапсуляція. Передача об'єктів методам класу. Правила використання модифікаторів get- та out- для параметрів методів.	10
	Разом	100

8. Індивідуальне навчально-дослідне завдання

Індивідуальне науково-дослідне завдання складається з двох напрямів:

I – підготовка та захист контрольної роботи (для студентів ЗФН);

II – підготовка мультипрезентації одного з напрямів сучасних технологій програмування (для студентів ДФН).

Підготовка та захист контрольної роботи:

Основне завдання цього виду діяльності – набуття практичних навичок самостійної розробки професійного програмного забезпечення і використання сучасних інформаційних технологій для розв'язання різноманітних задач у практичній діяльності.

Загальні вимоги до виконання індивідуального завдання:

- 1) Загальна характеристика об'єкта дослідження
- 2) Визначення потреб потенційних споживачів продукції.
- 3) Розробка ПП.
- 4) Оцінка якості продукції.

9. Форми роботи та критерії оцінювання

Рейтинговий контроль знань студентів здійснюється за 100-бальною шкалою:

Шкала оцінювання: національна та ECTS

ОЦІНКА ECTS	СУМА БАЛІВ	ОЦІНКА ЗА НАЦІОНАЛЬНОЮ ШКАЛОЮ	
		екзамен	залік
A	90-100	5 (відмінно)	5/відм./зараховано
B	80-89	4 (добре)	4/добре/ зараховано

C	65-79		
D	55-64	3 (задовільно)	3/задов./ зараховано
E	50-54		
FX	35-49	2 (незадовільно)	Не зараховано

Форми поточного та підсумкового контролю. Комплексна діагностика знань, умінь і навичок студентів із дисципліни здійснюється на основі результатів проведення поточного й підсумкового контролю знань (КР). Поточне оцінювання (індивідуальне, групове і фронтальне опитування, самостійна робота, самоконтроль). Завданням поточного контролю є систематична перевірка розуміння та засвоєння програмового матеріалу, виконання практичних, лабораторних робіт, умінь самостійно опрацьовувати тексти, складання конспекту рекомендованої літератури, написання і захист реферату, здатності публічно чи письмово представляти певний матеріал.

Завданням підсумкового контролю (КР, залік) є перевірка глибини засвоєння студентом програмового матеріалу модуля.

Критерії оцінювання відповідей на практичних заняттях:

Студенту виставляється відмінно, якщо студент здатний самостійно здійснювати основні види навчальної діяльності. Знання студента є глибокими, міцними, узагальненими; студент вміє застосовувати знання творчо, його навчальна діяльність позначена вмінням самостійно оцінювати різноманітні життєві ситуації, явища, факти, виявляти і відстоювати особисту позицію.

Студенту виставляється дуже добре, якщо студент знає істотні ознаки понять, явищ, закономірностей, зв'язків між ними, а також самостійно застосовує знання в нестандартних ситуаціях, володіє розумовими операціями, вміє робити висновки, виправляти допущені помилки. Відповідь повна, правильна, логічна, обґрунтована.

Студенту виставляється добре, якщо студент знає ознаки понять, явищ, закономірностей, зв'язків між ними на середньому рівні, а також самостійно застосовує знання в стандартних ситуаціях, володіє розумовими операціями, вміє робити висновки, виправляти допущені помилки. Відповідь повна, правильна, логічна, обґрунтована.

Студенту виставляється достатньо, якщо відповідь студента при відтворенні навчального матеріалу елементарна, зумовлюється початковими уявленнями про предмет вивчення. Студент відтворює основний навчальний матеріал, здатний виконувати завдання за зразком, володіє елементарними вміннями навчальної діяльності.

Студенту виставляється мінімальний задовільно, якщо відповідь студента при відтворенні навчального матеріалу елементарна, зумовлюється початковими уявленнями про предмет вивчення. Студент відтворює основний навчальний матеріал.

Кількість балів у кінці **семестру** повинна складати від 250 до 500 балів (за 5 кредитів), тобто сума балів за виконання усіх завдань.

Відповідний розподіл балів, які отримують студенти за 5 крд

Поточне оцінювання та самостійна робота														КР	Накопичувальні бали/ Сума
T1	T2	T3	T4	T5,6	T7	T8	T9	T10	T11	T12	T13	T14	T15		
30	35	35	10	40	30	35	35	10	20	20	30	35	35	100	500/100*

***Примітка.** Коефіцієнт для іспиту – 0,6. Іспит оцінюється в 40 б.

10. Засоби діагностики

Засобами діагностики та методами демонстрування результатів навчання є: завдання до лабораторних занять, завдання для самостійної та індивідуальної роботи, презентації результатів досліджень, тестові завдання, контрольні роботи.

11. Методи навчання

Усний виклад матеріалу: наукова розповідь, спрямована на аналіз фактичного матеріалу; пояснення – вербальний метод навчання, за допомогою якого розкривається сутність певного явища, закону, процесу; проблемне навчання, робота з підручником та додатковими джерелами.

Лекційні заняття призначені для теоретичного осмислення і узагальнення складних розділів курсу, які освітлюються, в основному, на проблемному рівні та у формі діалогічно-проблемних лекцій.

Лабораторні заняття є аудиторними, проводяться по наперед відомих темах у вигляді активних форми проведення занять. Вони призначені для закріплення і глибшого вивчення певних аспектів лекційного матеріалу на практиці.

Самостійна робота є позааудиторною і призначена для самостійного ознайомлення студента з певними розділами курсу за рекомендованими педагогом матеріалами і підготовки до виконання індивідуальних завдань по курсу.

Поточний рейтинг-контроль проводиться викладачем в процесі проведення всіх видів занять. Проміжний рейтинг-контроль призначений для практичної комплексної оцінки освоєння розділів курсу і здійснюється шляхом підготовки студентами відповідей на поставлені питання.

Кінцевий контроль знань з дисципліни проводиться під час складання іспиту.

12. Рекомендована література

Базова

1. Інформатика: візуальне програмування: навчально-методичний посібник/ Булгакова О.С., Зосімов В.В., Погромська Г.С., Махровська Н.А. – Херсон: ОЛДІ-ПЛЮС, 2014.- 312 с. (гриф МОН України, лист №1/11-1208 від 29.01.14)
2. Г. Шилдт. Полный справочник по С#. – М.: Издательский дом "Вильямс", 2008.
3. Э. Троелсен. С# и платформа .NET. Библиотека программиста. – СПб. : Питер, 2007.
4. Т.П. Караванова. Основи алгоритмізації та програмування. 750 задач з рекомендаціями та прикладами. – К.: Форум, 2002.
5. Э. Кингсли-Хьюджес, К. Кингсли-Хьюджес. С# 2005. Справочник программиста. – М.: ООО «ИД Вильямс», 2007.
6. Б. Керниган, Р. Пайк. Практика программирования. – СПб.: «Невский диалект», 2001.
7. В. О. Грязнова, С. В. Єфіменко. Основи методології програмування. – К.: ВПЦ «Київський університет», 2005.

Допоміжна

1. Zosimov V. Khrystodorov O., Bulgakova O. Dynamically changing user interfaces: software solutions based on automatically collected user information// Programming and Computer Software, vol 44 (6), 2018, P. 492-498.

2. МакГрат, М. Программирование на С для начинающих / М. МакГрат. - М.: Эксмо, 2015. - 192 с.
3. Перри, Г. Программирование на С для начинающих / Г. Перри, Д. Миллер. - М.: Эксмо, 2015. - 368 с.
4. Перри, Г. Программирование на С для начинающих / Г. Перри, Д. Миллер. - М.: Эксмо, 2016. - 192 с.
5. Васильев, А.Н. Программирование на С# для начинающих. Особенности языка / А.Н. Васильев. - М.: Эксмо, 2017. - 128 с.
6. Базурін В.М. Середовища програмування як засіб навчання учнів основ програмування / В. М. Базурін // Інформаційні технології і засоби навчання. - 2017. - Т. 59, вип. 3. - С. 13-27.
7. Герб Саттер, Андрій Александреску Стандарты программирования на С++,; Диалектика, 2016,224стр.
8. Algebra-Algorithmic Models and Methods of Parallel Programming; NAS of Ukraine, Institute of Software Systems / P.I. Andon, A.Yu. Doroshenko, K.A. Zhereb, O.A. Yatsenko. – Kyiv : Akadem- periodyka, 2018.- 192 p.
9. Prusov V., Doroshenko A. Computational techniques for modeling atmospheric processes.– Hershey, USA: IGI Global, 2018. – 460 p.
10. Тетяна Павловська С/С++.Программирование на языке высокого уровня: Учебник для вузов,-2016, 464стр.
11. Стенлі Ліпман, Жозі Лажойс, Барбара Му Язык программирования С++. Базовый курс ,- Диалектика, 2016, 1120стр
12. Седжвик Р. Алгоритмы на С++ 2-е изд. - М.: Интуит, 2016. - 1000 с.
13. Страуструп Бьярне. Программирование. Принципы и практика с использованием С++2-е издание. — Вильямс, 2015. — 1328 с.
14. Бьярне Страуструп Язык программирования С++. Краткий курс: Диалектика, 2019,320стр.
15. Доусон М. Изучаем С++ через программирование игр: Питер, 2016,352стр.
16. Алгоритмы. Справочник с примерами на С, С++, Java и Python, 2-е изд.: Пер. с англ. – СПб.: ООО “Альфа-книга”, 2017. – 432 с. : ил. 14. Златопольский Д.М. Основы программирования на языке Python. – М.: ДМК Пресс, 2018. – 284 с.
17. Петров В.Ю. Информатика. Алгоритмизация и программирование. Часть 1: Учебное пособие. - Санкт-Петербург: Университет ИТМО, 2016. - 91 с.
18. Лебедев И.С., Петров В.Ю. Информатика. Программирование. Часть 2: Учебнометодическое пособие. - Санкт-Петербург: Университет ИТМО, 2016. - 71 с.
19. Осипов Н.А. Технологии программирования: Учебное пособие. - СанктПетербург: Университет ИТМО, 2016. - 61 с.
20. Поляков К. Ю. Программирование. Python. С++. Часть 1. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 144 с.
21. Поляков К. Ю. Программирование. Python. С++. Часть 2. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 176 с.
22. Поляков К. Ю. Программирование. Python. С++. Часть 3. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 208 с.
23. Поляков К. Ю. Программирование. Python. С++. Часть 4. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 192 с.
24. Голиков Д.И. «Scratch для юных программистов», «БХВ-Петербург», Санкт-Петербург, 2017.
25. Язык С++ и программирование на нём: учебное пособие / В.И. Рейзлин; – 2-е изд., перераб. Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2015 – 212 с.
26. Т.А. Павловская. С/С++. Процедурное и объектноориентированное программирование. СПб: Питер, 2015. – 495с.

27. Б. Керниган, Д. Ритчи. Язык программирования Си. 2-е издание. – М.: Вильямс , 2015. – 304 с.
28. Richard L. Halterman Fundamentals of Programming: School of Computing Southern Adventist University, 2019, 775стр
29. Доусон М. Изучаем C++ через программирование игр: Питер СПб, 2016, 352стр.
30. Шлее М. Qt 5.3. Профессиональное программирование на C++. – БХВ-Петербург, 2015 – 929 с.
31. Боголюбов С.К. Объектно-ориентированное программирование в C++. – Питер, 2015. – 928 с.
32. Дашнер, С. Изучаем Java EE. Современное программирование для больших предприятий / С. Дашнер. - СПб.: Питер, 2015. - 94 с.
33. Дашнер, С. Изучаем Java EE. Современное программирование для больших предприятий / С. Дашнер. - СПб.: Питер, 2018. - 384 с.
34. Глушаков С.В., Дуравкина Т.В. Программирование на C++. – М.: АСТ, 2018. – 585 с.

13. Інформаційні ресурси

11. <https://dotnetfiddle.net/>
12. <https://rextester.com/>

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МИКОЛАЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ В. О. СУХОМЛИНСЬКОГО**

Механіко-математичний факультет

Кафедра комп'ютерних наук та прикладної математики



ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ПРОГРАМУВАННЯ

Ступінь бакалавра

Галузь знань 11 Математика та статистика
спеціальність 113 «Прикладна математика»
освітня програма Інформатика

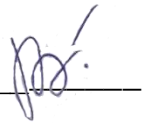
Програму розроблено та внесено: Миколаївський національний університет імені В. О. Сухомлинського

РОЗРОБНИК ПРОГРАМИ: Булгакова Олександра Сергіївна, доцент кафедри комп'ютерних наук та прикладної математики, кандидат технічних наук, доцент.

Програму схвалено на засіданні кафедри комп'ютерних наук та прикладної математики

Протокол від «27» серпня 2019 року № 1

Завідувач кафедри комп'ютерних наук та прикладної математики _____
(Поздєєв В.О.)



Програму погоджено навчально-методичною комісією механіко-математичного факультету

Протокол від «27» серпня 2019 року № _____
Голова навчально-методичної комісії _____ (Пархоменко О.Ю.)



Програму погоджено навчально-методичною комісією університету

Протокол від «27» серпня 2019 року № 14
Голова навчально-методичної комісії університету _____ (Кузнецова О.А.)



Програма вивчення нормативної навчальної дисципліни «Програмування» складена Булгаковою О.С. відповідно до ступеня «бакалавр» студентів спеціальності 113 Прикладна математика

Предметом вивчення навчальної дисципліни є вивчення основ алгоритмізації та побудови алгоритмів програм; вивчення базового синтаксису та основних елементів мови С#; вивчення основ ООП (об'єктно-орієнтованого програмування); створення консольних прикладень; створення простих Windows-додатків.

Міждисциплінарні зв'язки: структура курсу спрямована на постійну демонстрацію взаємозв'язку дисциплін, таких як «Алгоритми та структури даних», «ООП», та інші.

1. Мета та завдання навчальної дисципліни

Мета курсу – ознайомлення студентів з сучасною мовою програмування С# та оволодіння основними можливостями цієї мови, навичками хорошого стилю програмування, методами проектування та створення програм згідно сучасних технологій програмування.

Завдання дисципліни:

Студент повинен знати:

основні елементи мови С# – типи даних, оператори та керуючі інструкції мови; основні принципи ООП, правила визначення класів в мові С#.

Студент повинен вміти:

побудувати алгоритм розв'язку задачі відповідної складності, порівняти різні алгоритми і обрати оптимальний для конкретної задачі, створити консольне прикладення, що реалізує обраний алгоритм; налагодити та протестувати програму; використовуючи панель інструментів середовища, створити просте Windows-прикладення.

Згідно з вимогами освітньо-професійної програми студент оволодіває такими компетентностями:

I. Загальнопредметні:

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК06. Здатність до абстрактного мислення, аналізу та синтезу

II. Фахові:

ФК07. Здатність експлуатувати та обслуговувати програмне забезпечення автоматизованих та інформаційних систем різного призначення.

ФК08. Здатність використовувати сучасні технології програмування та тестування програмного забезпечення.

На вивчення навчальної дисципліни відводиться 150 годин/5 кредитів ECTS.

2. Інформаційний обсяг навчальної дисципліни

Кредит 1. Базові поняття програмування.

Тема 1. Основні поняття програмного забезпечення. Огляд процесу розробки програмного забезпечення. Етапи виконання комп'ютерної програми. Блок-схема та псевдокод як засоби, що використовуються у процесі розробки програми. Мови програмування. Процедурні та об'єктно-орієнтовані мови програмування. Місце мови С# серед сучасних мов програмування.

Тема 2. Представлення даних та двійкова арифметика. Необхідність типізації даних. Фундаментальні поняття типів даних та змінних. Змінні та константи. Правила іменування змінних. Операції. Використання програмного синтаксису.

Основні типи конструкцій в програмуванні. Використання розгалужень та циклів. Способи визначення логічних помилок. Культура програмування, поняття про ефективність алгоритму.

Тема 3. Процедури та функції як засіб створення модульних програм. Поняття аргументу, поняття параметра. Формальні і фактичні параметри, їх зв'язок за значенням та за адресою. Вхідні та вихідні параметри. Повернення скалярного значення функцією. Рекурсія.

Кредит 2. Загальний огляд платформи Microsoft .NET

Тема 4. Огляд платформи Microsoft .NET. Огляд циклу розробки програмного забезпечення. Переваги платформи .NET.

Тема 5. Компоненти платформи .NET. Формат виконуваних програмних модулів (EXE і DLL), загальномовне виконуюча среда, Common Language Runtime, CLR, набір програмних засобів.

Кредит 3. Основи мови програмування C#

Тема 6. Структура програми мовою C#. Основні елементи програми. Основні операції вводу/виводу. Засоби форматування. Компілювання, виконання та відлагодження.

Тема 7. Огляд загальної системи типів. Використання вбудованих типів-значень даних у мові C#. Створення типів даних користувача в мові C# (злічені типи та структури). Перетворення типів (явне та неявне). Використання пробілів та дужок у виразах для покращення читання програми.

Тема 8. Використання основних програмних конструкцій у мові C#. Розгалуження. Цикли. Поняття виключної ситуації. Обробка основних виключних ситуацій. Генерування виключних ситуацій.

Кредит 4. Поняття про класи та методи. Робота з масивами даних в мові C#

Тема 9. C# як об'єктно-орієнтована мова. Поняття про визначення класу та створення об'єкту. Типи-значення та типи-посилання. Визначення класу, доступ до даних-членів класу. Визначення методів класу.

Тема 10. Використання методів. Використання параметрів. Параметри за замовчуванням. Рекурсія.

Тема 11. Загальний огляд масивів як способу групування даних. Створення та використання масивів у мові C#. Використання конструкції new. Багатовимірні масиви. Масиви як параметри методів. Методи, що повертають масиви. Цикл по елементах масива (foreach).

Кредит 5. Використання класів в мові C#

Тема 12. Створення об'єкту даного класу. Конструктор класу. Конструктор за замовчуванням та конструктор з параметрами. Створення параметризованих об'єктів класу.

Тема 13. Деструктор класу. Поняття про систему збору «сміття». Використання службового слова this.

Тема 14. Управління доступом до членів класу, інкапсуляція. Передача об'єктів методам класу. Правила використання модифікаторів ref- та out- для параметрів методів.

3. Рекомендована література Базова

1. Г. Шилдт. Полный справочник по C#. – М.: Издательский дом "Вильямс", 2008.
2. Э. Троелсен. C# и платформа .NET. Библиотека программиста. – СПб. : Питер, 2007.
3. Т.П. Караванова. Основи алгоритмізації та програмування. 750 задач з рекомендаціями та прикладами. – К.: Форум, 2002.
4. Э. Кингсли-Хьюджес, К. Кингсли-Хьюджес. C# 2005. Справочник программиста. – М.:

ООО «ИД Вильямс», 2007.

5. Б. Керниган, Р. Пайк. Практика программирования. – СПб.: «Невский диалект», 2001.
6. В. О. Грязнова, С. В. Єфіменко. Основы методологии програмування. – К.: ВПЦ «Київський університет», 2005.

Допоміжна

1. Zosimov V. Khrystodorov O., Bulgakova O. Dynamically changing user interfaces: software solutions based on automatically collected user information// Programming and Computer Software, vol 44 (6), 2018, P. 492-498.
2. МакГрат, М. Программирование на С для начинающих / М. МакГрат. - М.: Эксмо, 2015. - 192 с.
3. Перри, Г. Программирование на С для начинающих / Г. Перри, Д. Миллер. - М.: Эксмо, 2015. - 368 с.
4. Перри, Г. Программирование на С для начинающих / Г. Перри, Д. Миллер. - М.: Эксмо, 2016. - 192 с.
5. Васильев, А.Н. Программирование на С# для начинающих. Особенности языка / А.Н. Васильев. - М.: Эксмо, 2017. - 128 с.
6. Базурін В.М. Середовища програмування як засіб навчання учнів основ програмування / В. М. Базурін // Інформаційні технології і засоби навчання. - 2017. - Т. 59, вип. 3. - С. 13-27.
7. Герб Саттер, Андрій Александреску Стандарты программирования на С++,; Диалектика, 2016, 224стр.
8. Algebra-Algorithmic Models and Methods of Parallel Programming; NAS of Ukraine, Institute of Software Systems / P.I. Andon, A.Yu. Doroshenko, K.A. Zhreb, O.A. Yatsenko. – Kyiv : Akadem- periodyka, 2018.- 192 p.
9. Prusov V., Doroshenko A. Computational techniques for modeling atmospheric processes.– Hershey, USA: IGI Global, 2018. – 460 p.
10. Тетяна Павловська С/С++.Программирование на языке высокого уровня: Учебник для вузов,-2016, 464стр.
11. Стенлі Ліппман, Жозі Лажойс, Барбара Му Язык программирования С++. Базовый курс ,- Диалектика, 2016, 1120стр
12. Седжвик Р. Алгоритмы на С++ 2-е изд. - М.: Интуит, 2016. - 1000 с.
13. Страуструп Бьярне. Программирование. Принципы и практика с использованием С++2-е издание. — Вильямс, 2015. — 1328 с.
14. Бьярне Страуструп Язык программирования С++. Краткий курс: Диалектика, 2019, 320стр.
15. Доусон М. Изучаем С++ через программирование игр: Питер, 2016, 352стр.
16. Алгоритмы. Справочник с примерами на С, С++, Java и Python, 2-е изд.: Пер. с англ. – СПб.: ООО “Альфа-книга”, 2017. – 432 с. : ил. 14. Златопольский Д.М. Основы программирования на языке Python. – М.: ДМК Пресс, 2018. – 284 с.
17. Петров В.Ю. Информатика. Алгоритмизация и программирование. Часть 1: Учебное пособие. - Санкт-Петербург: Университет ИТМО, 2016. - 91 с.
18. Лебедев И.С., Петров В.Ю. Информатика. Программирование. Часть 2: Учебнометодическое пособие. - Санкт-Петербург: Университет ИТМО, 2016. - 71 с.
19. Осипов Н.А. Технологии программирования: Учебное пособие. - СанктПетербург: Университет ИТМО, 2016. - 61 с.
20. Поляков К. Ю. Программирование. Python. С++. Часть 1. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 144 с.

21. Поляков К. Ю. Программирование. Python. C++. Часть 2. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 176 с.
22. Поляков К. Ю. Программирование. Python. C++. Часть 3. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 208 с.
23. Поляков К. Ю. Программирование. Python. C++. Часть 4. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019. – 192 с.
24. Голиков Д.И. «Scratch для юных программистов», «БХВ-Петербург», Санкт-Петербург, 2017.
25. Язык C++ и программирование на нём: учебное пособие / В.И. Рейзлин; – 2-е изд., перераб. Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2015 – 212 с.
26. Т.А. Павловская. C/C++. Процедурное и объектноориентированное программирование. СПб: Питер, 2015. – 495с.
27. Б. Керниган, Д. Ритчи. Язык программирования Си. 2-е издание. – М.: Вильямс , 2015. – 304 с.
28. Richard L. Halterman Fundamentals of Programming: School of Computing Southern Adventist University, 2019, 775стр
29. Доусон М. Изучаем C++ через программирование игр: Питер СПб, 2016, 352стр.
30. Шлее М. Qt 5.3. Профессиональное программирование на C++. – БХВ-Петербург, 2015 – 929 с.
31. Боголюбов С.К. Объектно-ориентированное программирование в C++. – Питер, 2015. – 928 с.
32. Дашнер, С. Изучаем Java EE. Современное программирование для больших предприятий / С. Дашнер. - СПб.: Питер, 2015. - 94 с.
33. Дашнер, С. Изучаем Java EE. Современное программирование для больших предприятий / С. Дашнер. - СПб.: Питер, 2018. - 384 с.
34. Глушаков С.В., Дуравкина Т.В. Программирование на C++. – М.: АСТ, 2018. – 585 с.

4. Форма підсумкового контролю успішності навчання: іспит

5. Засоби діагностики успішності навчання:

Лекційні та лабораторні заняття з використанням традиційних методів, інтерактивних методик та комп'ютерної техніки, індивідуальна робота, самостійна робота.

Поточний та підсумковий контроль здійснюється у вигляді комп'ютерних тестів на освітньому просторі університету. Для оцінювання використовується національна чотирьохбальна шкала: відмінно, добре, задовільно, незадовільно; європейська шкала: А, В, С, D, E, FX, F.

(приклад для заліку) 100% балів студенти накопичують на заняттях та під час поточного і підсумкового контролю, що регламентується робочою програмою викладача. (приклад для іспиту) 60% балів студенти накопичують на заняттях та під час поточного контролю, що регламентується робочою програмою викладача, 40% балів студенти набирають на іспиті.