


Уманський державний педагогічний університет імені Павла Тичини  
Кафедра інформатики і інформаційно-комунікаційних технологій

**“ЗАТВЕРДЖУЮ”**

Завідувач кафедри

 М. О. Медведєва

“ 17 ” 08 2019 року

## ***РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ***

### **ВВС3.2.03 Мови програмування**

Спеціальність 014.06 Середня освіта (Хімія)

Освітня програма Середня освіта (Хімія)

Природничо-географічний факультет

2019 – 2020 навчальний рік


Робоча програма навчальної дисципліни «Мови програмування» для студентів спеціальності 014.06 Середня освіта (Хімія).

Розробники: ст. викладач кафедри інформатики і інформаційно-комунікаційних технологій к.т.н. Л. О. Коршевнюк

Робочу програму схвалено на засіданні кафедри інформатики і інформаційно-комунікаційних технологій

Протокол № 1 від “27” серпня 2019 року

Завідувач кафедри  
інформатики і інформаційно-  
комунікаційних технологій

  
\_\_\_\_\_ М. О. Медведєва

Робочу програму розглянуто та затверджено на засіданні науково-методичної комісії природничо-географічного факультету

Протокол № 1 від “29” серпня 2019 року

Голова науково-методичної комісії природничо-географічного факультету

  
\_\_\_\_\_ С. Л. Грабовська

## 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітній ступінь	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 5	Галузь знань <i>01 Освіта</i>	Вибіркова	
Модулів – 1	Спеціальність 014.06 Середня освіта (Хімія) Предметна спеціалізація 014.09 Середня освіта (Інформатика)	Рік підготовки	
Змістових модулів – 2		2-й	-й
Індивідуальне науково-дослідне завдання  _____ (назва)		Семестр	
Загальна кількість годин – 150		3-й	-й
Тижневих годин для денної форми навчання: аудиторних – 5 самостійної роботи студента – 5	Освітній ступінь: магістр	Лекції	
		32 год.	год.
		Практичні, семінарські	
		год.	год.
		Лабораторні	
		44 год.	год.
		Самостійна робота	
		74 год.	год.
		Індивідуальні завдання:	
		год.	год.
Вид контролю:			
залік			

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить:

для денної форми навчання – 1/3

## 2. Мета та завдання навчальної дисципліни

**Мета** курсу полягає у поглибленні фундаментальних понять, методів та ідіом програмування: побудови алгоритмів, структур даних, вибору інструментів розробки комп'ютерної програми, мови програмування, методології і технології програмування та методів їх застосування для розв'язання різних класів задач, а також в ознайомленні магістрів сучасними підходами до програмування, оцінкою якості програмного забезпечення, вимогами до програмного коду, що повторно використовується; шляхами використання шаблонів проектування; виробленні навичок розробки програмних систем з використанням наскрізного прикладу еволюції програмного додатку.

### **Завдання:**

Вивчення цієї дисципліни дає можливість студенту одержати різнобічні знання про сучасний стан і тенденції розвитку мов програмування, їх використання для розробки додатків, важливі складові сучасних інформаційних технологій, таких, як графічні процесори, електронні таблиці, системи управління базами даних, системи програмування, операційні системи й оболонки, а також набуття студентами:

- уміння здійснювати постановку, формалізацію й алгоритмізацію економічних і управлінських завдань, їх рішення на ПЕОМ, як за допомогою алгоритмічних мов програмування, так і за допомогою систем управління базами даних та засобів автоматизації розробки програмних систем;
- навичок практичної розробки програмного забезпечення, призначеного для вирішення прикладних завдань.
- забезпечити ґрунтовне вивчення студентами тих понять і методів мов програмування, які можуть бути використанні ними в процесі навчання окремих тем шкільного курсу інформатики, проведенні факультативних занять в закладах освіти;
- сформувані у студентів достатні знання, вміння і навички, необхідні для практичного проведення навчально-виховної роботи в основній та старшій школі (профільне навчання) з широким використанням засобів сучасних інформаційних технологій.
- Згідно з вимогами освітньо-професійної програми студенти повинні володіти компетентностями:
- **ФК 24.** Володіє теоретичними основами комп'ютерної графіки, засобами та методами введення, обробки, конвертації і виведення графічної інформації за допомогою пакетів графічних програм та навичками практичної роботи з графічними та мультимедійними пакетами.
- **ФК 26.** Здатність формувати технічно, технологічно, інформаційно-освічену особистість, підготовлену до активної трудової діяльності в умовах сучасного високотехнологічного інформаційного суспільства.

### **Очікувані результати навчання**

В результаті вивчення навчальної дисципліни студент повинен

**знати:** загальну класифікацію програмного забезпечення та мов програмування; основні парадигми програмування; синтаксис таких мов програмування як C/C++, PHP, Java, Assembler, SQL та структуру робочого середовища таких компіляторів як C++ Builder, Visual Studio, .Net та C#, Java тощо; основні методи та засоби обробки текстів, масивів, складних структур даних, баз даних.

**уміти:** обирати мову програмування, що найкраще підходить для розв'язку задач певного класу, розробляти та реалізувати програмне забезпечення для вирішення класу задач; реалізувати програми у середовищі різних середовищах; розробляти програми в діалоговому та інтерактивному режимах; створювати структури даних для опису об'єктів;

розробляти програмні структури за принципами об'єктно-орієнтованого, ієрархічного, модульного та компонентного програмування тощо.

. Згідно з вимогами освітньо-професійної програми студенти повинні володіти програмними результатами навчання:

**ПРН 32.** Володіє комп'ютерними методами аналізу та обробки інформації, здатний використовувати ці результати у професійній діяльності.

### **3. Мова навчання:**

Мова навчання: українська

## **4. Програма навчальної дисципліни**

### **Змістовий модуль 1. Теоретичні засади мов програмування**

Тема 1. Визначення мови програмування

Тема 2. Особливості застосування мов програмування у різних сферах

Тема 3. Критерії ефективності мов програмування

Тема 4. Віртуальні машини та трансляція мов

Тема 5. Види мов програмування

Тема 6. Вирази та присвоювання в мовах програмування дії та оператори

Тема 7. Засоби представлення синтаксису мов

Тема 8. Формальна семантика мов

### **Змістовий модуль 2. Особливості мов програмування**

Тема 1. Особливості об'єктно-орієнтованих мов

Тема 2. Основні поняття паралельного програмування

Тема 3. Функціональне програмування

Тема 4. Логічне програмування

Тема 5. Мови для роботи з базами даних

Тема 6. Мови для автоматизації розробки програмних продуктів

Тема 7. Навчальні мови програмування

Тема 8. Мови імітаційного моделювання



## 6. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1.	Структурне програмування	2
2.	Конструкції мови, дії та оператори	2
3.	Синтаксичні конструкції мов програмування	2
4.	Лексеми та лексичний синтаксис	2
5.	Граматики в мовах програмування	2
6.	Обробка асоціативності та пріоритетності	2
7.	Синтаксичні схеми	2
8.	Формальна семантика мов програмування	2
9.	Умовні оператори та цикли	2
10.	Типізація даних	2
11.	Скалярні та складені типи	2
12.	Організація підпрограм	2
13.	Абстрактні типи даних	2
14.	Об'єктно-орієнтоване програмування	2
15.	Аспектно-орієнтоване програмування	2
16.	Апарат виключень	2
17.	Введення-виведення в різних мовах програмування	2
18.	Процеси та потоки	2
19.	Особливості використання мов роботи з базами даних	2
20.	Мови, призначені для розробки Web-додатків	2
21.	Мови, які використовуються на браузері клієнта	2
22.	Мова UML	2
23.	ВСЬОГО	44

## 7. Самостійна робота

№ з/п	Назва теми	Кількість годин
1.	Розвиток стилів та мов програмування	2
2.	Сучасні інтегровані середовища розробки	4
3.	Синтаксичні схеми	4
4.	Формальна семантика мов програмування	4
5.	Умовні оператори та цикли	4
6.	Типізація даних	4
7.	Скалярні та складені типи	4
8.	Організація підпрограм	4
9.	Абстрактні типи даних	2
10.	Об'єктно-орієнтоване програмування. Особливості мов.	4
11.	Мови, призначені для розробки додатків, що виконуються на сервері для підтримки Web-сайту	4
12.	Мови, які використовуються на браузері (програми перегляду) клієнта JavaScript, JScript, VBScript.	4
13.	Мови штучного інтелекту	4
14.	Мови логічного програмування	4
15.	Мови SDL	4
16.	Спеціальні мови програмування у комп'ютерних математичних системах	4
17.	Використання мови LOGO	2
18.	Kid's Programming Language	4

19.	Мова GPSS	4
20.	Мова UML	4
	<b>ВСЬОГО</b>	<b>74</b>

## 8. Методи навчання

1. Пояснювально-ілюстративний метод або інформаційно-рецептивний.
2. Репродуктивний метод (репродукція - відтворення).
3. Частково-пошуковий, або евристичний метод.
4. Метод проектів.
5. Неімітаційні методи - способи активізації пізнавальної діяльності на лекційних заняттях.

Основними видами занять, які проводяться під керівництвом викладача, є лекції та лабораторні роботи і самостійна робота. На лекціях розглядаються загальні теоретичні положення дисципліни. Під час проведення лекцій використовуються мультимедійні засоби для інтерактивної демонстрації прикладів та графічного матеріалу. До кожної лекції студентам додається презентація основних положень. При виконанні лабораторних робіт зміцнюються знання, отримані на лекціях, набуваються первинні навички з організації та обслуговування баз даних, розглядаються типові предметні області, дії розробників та адміністраторів в стандартних ситуаціях. При самостійній роботі студенти набувають навички програмування з використанням мови C++, які не використані в навчальному процесі та поглиблюються свої знання щодо технологій доступу до даних. При проведенні лекційних та практичних занять використовується інструментальне середовище розробки MS Visual Studio.

Самостійна робота є позааудиторною і призначена для самостійного ознайомлення студента з певними розділами курсу за рекомендованими педагогом матеріалами і підготовки до виконання самостійних завдань по курсу. Необхідним елементом успішного засвоєння навчального матеріалу дисципліни є самостійна робота студентів з вітчизняною та закордонною спеціальною літературою. Самостійна робота є основним засобом оволодіння навчальним матеріалом у час, вільний від обов'язкових навчальних занять. Самостійна робота студентів передбачає поглиблене вивчення тем з використанням рекомендованої літератури, пошук інформації в Інтернеті, а також додаткову роботу в комп'ютерних класах для виконання завдань. Домашні завдання дослідницького і творчого характеру передбачають ознайомлення студентів з більш розширеним розглядом матеріалом теоретичного плану ніж той, що передбачено на лекціях та придбання навичок програмування.

## 9. Методи контролю

Контроль знань і умінь студентів (поточний і підсумковий) з дисципліни здійснюють згідно з кредитно-модульною системою організації навчального процесу.

Система контролю над самостійною роботою студентів включає:

а) опитування студентів по змісту теоретичних знань під час проведення лабораторних занять;

б) перевірка виконання самостійних робіт, контрольних робіт;

в) захист індивідуальних завдань.

**Поточний** – під час виконання практичних (семінарських), лабораторних робіт, індивідуальних завдань (описових робіт (ОР), розрахунково-графічних робіт (РГР), розрахункових робіт (РР), написання рефератів), контроль за засвоєння певного модуля (модульний контроль). Форму проведення поточного контролю і систему оцінювання визначає відповідна кафедра.

**Підсумковий** – включає диференційований залік і екзамен.



Якщо навчальна дисципліна складається з **3-4** модулів, то кожен модуль може оцінюватись в умовних балах пропорційно обсягу часу, відведеному на засвоєння матеріалу цього модуля. Максимально можлива кількість умовних балів за **навчальні заняття** студента становить 70% (коефіцієнт 0,7) і 30% (коефіцієнт 0,3) припадає на **екзамен (або диференційований залік)** від загальної кількості умовних балів.

Рішенням кафедри за виконання робіт, які не передбачені навчальним планом, але сприяють підвищенню рівня знань та умінь студентів з навчальної дисципліни (доповідь на студентській конференції, здобуття призового місця на олімпіадах, виготовлення макетів, підготовка наочних посібників тощо) може надаватись до **10%** від загальної кількості умовних балів з навчальної дисципліни.

Під час вивчення дисципліни використовуються наступні методи контролю: поточне та підсумкове тестування за теоретичним матеріалом, захист лабораторних робіт у формі співбесіди, практичні контрольні за результатами вивчення теми. Залік проводиться у вигляді виконання практичного завдання та відповіді на теоретичні питання у формі співбесіди.

## **10.Критерії оцінювання результатів навчання**

При визначенні оцінки знань студентів викладач керується критеріями і положеннями, які розроблені Міністерством освіти України. Крім цього, при оцінюванні засвоєння знань студентів проводиться після вивчення модуля модульний контроль, результати, якого оцінюються за нижче наведеними критеріями та заносяться в академічний журнал окремим рядком.

Оцінки «відмінно» заслуговують відповіді студентів, що показали глибокі знання програми курсу, засвоїли основну літературу, набули визначеного програмою кола необхідних знань та мають визначені практичні навички. Крім того, враховується вміння грамотно, чітко і послідовно викласти питання, що розглядається. Швидко виконувати поставлені практичні задачі.

Оцінки «добре» заслуговують відповіді студентів, які виявили тверді знання навчального матеріалу в обсязі навчальної програми, не допустили суттєвих помилок при відповіді на всі питання, чітко виклали матеріал та показали здібності і вміння застосовувати одержані знання у своїй практичній діяльності при використанні натяків на виконання практичних завдань.

Оцінки «задовільно» виставляється студентам, які показали знання основного матеріалу в обсязі навчальної програми, але допустили при відповіді помилки, невірність окремих формулювань, положень, або які показали поверхові знання з питань, не чітко і не аргументовано дали відповідь на поставлені запитання. Виконувати практичні завдання можуть виконувати лише при додаткових вказівках.

Оцінки «незадовільно» заслуговують відповіді студентів, які показали недостатні знання навчально-програмового матеріалу, не знають основних понять та положень дисципліни, не вміють виконувати елементарні практичні завдання і зовсім не орієнтуються в питаннях до практичної роботи, чим виявили непридатність до професійної діяльності.

Контроль успішності студентів з врахуванням поточного і підсумкового оцінювання проводяться відповідно до навчально-методичної карти, де зазначено види і терміни контролю. Система рейтингових балів для різних видів контролю та порядок їх переводу до національної (4-х бальної) та європейської (ECTS) шкали представлені в таблицях нижче. Для обліку і реєстрації показників успішності використовується рейтингова відомість успішності студентів. Кожний кредит включає бал оцінки поточної роботи студента на лекційних та практичних заняттях і результати виконання самостійних та індивідуальних завдань і їх захисту. Рейтинговий бал встановлюється як сума усіх оцінок,

отриманих за усі види робіт даного кредиту. Виконання тестових завдань здійснюється в режимі комп'ютерної діагностики або за допомогою роздрукованих завдань. Оцінювання тестів проводиться за власною методикою з приведенням підсумку до встановленої в рейтинговій шкалі балів. Заходи з модульного контролю проводяться по завершенню вивчення навчального матеріалу кожного модуля. За роботу на занятті (лекціях, практичних) нараховуються додаткові бали за кожне заняття, а за пропуски навчальних занять (лекцій, практичних) з неповажних причин можуть нараховуватися штрафні бали (1 бал за кожен пропуск), які віднімаються від сумарного рейтингового балу при виведенні підсумкової оцінки. При згоді студента до відомості обліку успішності може бути проставлена залікова оцінка на підставі поточного рейтингового балу. Студент може підвищувати оцінку "задовільно" або "добре", яку він отримав за результатами поточного контролю, складанням семестрового заліку або екзамену. 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 або написанням підсумкових контрольних робіт.

## 11. Розподіл балів, які отримують студенти

### Приклад для заліку

Поточне тестування та самостійна робота, залік - 20																Сума
Змістовий модуль 1								Змістовий модуль 2								
T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	100
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	

T1, T2 ... T9 – теми змістових модулів.

### Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою	
	для екзамену, курсової роботи, практики	для заліку
90–100	відмінно	зараховано
82–89	добре	
75–81		
69–74		
60–68	задовільно	не зараховано з можливістю повторного складання
35–59	незадовільно з можливістю повторного складання	
1–34	незадовільно з обов'язковим повторним вивченням дисципліни	

## 12. Методичне забезпечення

Бібліотека університету має достатню кількість підручників та посібників, практичних практикумів та іншої літератури для забезпечення повноцінного навчання студентів. Крім

цього на кафедрі наявна бібліотека для користування студентами на електронних носіях. Конспекти лекцій, інструкції до практичних робіт та додаткові матеріали розміщуються на сервері кафедри та у навчально-методичному комплексі дисципліни.

### **13. Рекомендована література**

#### **Основна**

1. Computing Curricula 2005. The Overview Report. [Електронний ресурс]. – Режим доступу: [www.acm.org/education/curricula.html](http://www.acm.org/education/curricula.html).
2. Обеспечение систем обработки информации программное. Термины и определения. ГОСТ 19781-90. – [Чинний від 1992-02-01 до 2007-12-10] – 16 с.– (Міждержавний стандарт).
3. 3. Systems and software engineering – Software Life Cycle Processes. ISO 12207:2008. – [Чинний від 2008-02-01] – II, 122 с.– (Міжнародний стандарт).
4. Лафоре, Р. Объектно-ориентированное программирование в С++. Класика Computer Science: пер. с англ./ Р. Лафоре. СПб.: Питер, 2005. 924 с.
5. Шилдт, Г. Полный справочник по С++: пер. с англ./ Г. Шилдт. М.: Издательский дом "Вильямс", 2006. 800 с.
6. Шилдт, Г. С++. Руководство для начинающих. М.: Издательский дом "Вильямс", 2006. 672 с.
7. Страуструп, Б. Язык программирования С++. Специальное издание. М.: "Бином"- "Невский диалект", 2012. 1136 с.
8. Березин, В.И. Начальный курс С и С++. М.: Диалог-Мифы, 2003. 288 с.
9. Архангельский, А. Я. Программирование в С++ Builder 6. М.: Бином-Пресс, 2007. – 1184с.
10. Седжвик, Р. Фундаментальные алгоритмы на С. Анализ/ Структуры данных/ Сортировка/ Поиск. СПб: ООО "ДиаСофт", 2003. 672 с.
11. Солтер, Николас А. С++ для профессионалов. М.: Издательский дом "Вильямс", 2006. 912 с.
12. Терехов А.М. Вступ до технологій програмування. [Електронний ресурс]. Режим доступу: <http://www.intuit.ru/department/sc/introprogteach/2>.
13. 17. Oracle® Unified Method (OUM). [Електронний ресурс]. Режим доступу: [www.oracle.com/us/products/consulting/resource-library/oracle-unified-method-069204.pdf](http://www.oracle.com/us/products/consulting/resource-library/oracle-unified-method-069204.pdf).
14. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. СПб.: Изд-во «Питер», 2007. – 366 с.
15. 40. Buschmann F., Meunier R., Rohnert H., Sommerlad P., Stal M. Pattern-Oriented Software Architecture, Volume 1, A System of Patterns. Wiley, 1996. – 476 с.

#### **Допоміжна**

16. Элиенс А. Принципы объектно-ориентированной разработки программ. 2-е издание. – М.: Издательский дом «Вильямс», 2002. 496 с.
17. Йордон Э., Аргила К. «Структурные модели в объектно-ориентированном анализе и проектировании». Издательство «Лори», 1999
18. Шаммас Н.К. Основы С++ и объектно-ориентированного программирования. – К. : Диалектика, 1996. 448с.
19. Вирт Н. Алгоритмы и структуры данных. М.: Мир, 1989
20. Ахо, Хопкрофт, Ульман. Структуры данных и алгоритмы. М.: Вильямс, 2000. 384 с.

### **14. Інформаційні ресурси**

1. <http://www.youtube.com/watch?v=rbCkcBDobCY>
2. <http://sdb.su/oop/>

3. <https://courses.prometheus.org.ua/>
4. <http://younglinux.info/oopython.php>
5. <http://msdn.microsoft.com>
6. <http://www.weblibrary.biz/c-sharp/princypy>
7. <https://spalah.com/course/oop>
8. <http://proger.elitno.net/oop/>