

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»

Кафедра системного аналізу та обчислювальної математики
(найменування кафедри)

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Програмування

(назва навчальної дисципліни)

Освітня програма: інтелектуальні технології та прийняття рішень в складних системах
(назва освітньої програми)

Спеціальність: 124 Системний аналіз
(найменування спеціальності)

Галузь знань: 12 Інформаційні технології
(найменування галузі знань)

Ступінь вищої освіти: бакалаврський
(назва ступеня вищої освіти)

Затверджено на засіданні кафедри

СА та ОМ

(найменування кафедри)

Протокол № 18 від 16.08.2021 р.

м. Запоріжжя 2021

1. Загальна інформація	
Назва дисципліни	<i>Програмування</i>
Рівень вищої освіти	<i>Перший (бакалаврський) рівень</i>
Викладач	<i>Широкорад Дмитро Вікторович, к.ф.-м.н., ст. викладач</i>
Контактна інформація викладача	<i>Телефон кафедри 769-8247, Телефон викладача +380631182567, hoveringphoenix@gmail.com</i>
Час і місце проведення навчальної дисципліни	<i>Предметна аудиторія кафедри, комп'ютерні класи згідно розкладу занять</i>
Обсяг дисципліни	<i>розподіл годин: лекції-30, лабораторні заняття-14, вид контролю- екзамен.</i>
Консультації	<i>Згідно з графіком консультацій</i>
2. Пререквізити і постреквізити навчальної дисципліни	
<i>Пререквізити: основи математичного аналізу, лінійна алгебра.</i>	
3. Характеристика навчальної дисципліни	
<p>Загальні компетентності:</p> <ul style="list-style-type: none"> – K01. Здатність до абстрактного мислення, аналізу та синтезу; – K02. Здатність застосовувати знання у практичних ситуаціях – K04. Знання та розуміння предметної області та розуміння професійної діяльності; – K07. Здатність до пошуку, оброблення та аналізу інформації з різних джерел; – K10. Здатність працювати автономно; – K12. Здатність працювати в команді. <p>Фахові компетентності:</p> <ul style="list-style-type: none"> – K18. Здатність формалізувати проблеми, описані природною мовою, у тому числі за допомогою математичних методів, застосовувати загальні підходи до математичного моделювання конкретних процесів; – K22. Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, інтелектуального аналізу даних. <p>Результати навчання:</p> <ul style="list-style-type: none"> – ПР08. Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій. – ПР09. Вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень. <p>В результаті вивчення дисципліни «Програмування» студент повинен знати:</p> <ul style="list-style-type: none"> – основні принципи роботи алгоритмів; – основні принципи створення сучасних програмних продуктів; – сучасні мови програмування та сфери їхнього практичного застосування; – методи реалізації алгоритмів за допомогою мов програмування високого рівня <p>вміти:</p> <ul style="list-style-type: none"> – реалізовувати лінійні та нелінійні алгоритми за допомогою мов програмування високого рівня; – самостійно створювати прикладні програми для вирішення типових задач; – застосовувати здобуті навички для обміну інформацією між різними програмними продуктами 	
4. Мета вивчення навчальної дисципліни	
<p><i>Метою викладання дисципліни "Програмування" є:</i></p> <ul style="list-style-type: none"> • <i>вивчення студентами основних принципів алгоритмізації і програмування;</i> • <i>ознайомлення студентів з сучасними напрямками розвитку алгоритмічних методів;</i> • <i>ознайомлення студентів з сучасними алгоритмічними мовами програмування та їхніми принципами роботи;</i> • <i>набуття студентами практичних навичок створення алгоритмів для вирішення типових задач;</i> 	

- набуття студентами практичних навичок створення прикладних програм з використанням мови програмування високого рівня ;
- розвиток логічного мислення;
- розвиток технічної культури;
- розвиток аналітичного мислення.

5. Завдання вивчення дисципліни

Навчити студентів застосовувати сучасні мови програмування високого рівня, а також самостійно створювати власні прикладні програми.

6. Зміст навчальної дисципліни

У навчальній дисципліні «Програмування» розглядаються сучасні технології програмування за допомогою алгоритмічних мов на прикладі мови C++. Вивчаються принципи створення алгоритмів для розв'язку типових задач. Курс містить відомості про базові типи даних C++, основні синтаксичні конструкції, реалізацію функцій. Також розглядаються методи роботи з рядковими даними, динамічні структури та деякі стандартні бібліотеки.

7. План вивчення навчальної дисципліни

№ тижня	Назва теми	Форми організації навчання	Кількість годин
1.	Сучасні технології створення програмного забезпечення ЕОМ. Елементи сучасної технології створення програм і можливості сучасних мов та систем програмування. Структуровані, блочні, модульні програми. Рівні програмування.	Лекції, лабораторні заняття, самостійна робота	4
2	Базові типи даних. Лінійні алгоритми та алгоритми, що розгалужуються. Поняття інформаційної одиниці, типу інформаційної одиниці, об'єкту. Атрибути об'єкту: адреса, значення, ідентифікатор, тип. Ідентифікатори та їхні атрибути: тип, сфера дії, область видимості, тривалість, клас пам'яті, простір імен, тип компонування. Оператори розгалуження.	Лекції, лабораторні заняття, самостійна робота	6
3	Цикли та одновимірні масиви. Показники і масиви. Цикл з параметром. Оператори передачі управління break, continue, return, goto.	Лекції, лабораторні заняття, самостійна робота	4
4	Цикли з перед- та післяумовою. Структурні схеми операторів циклів. Цикли з передумовою та цикли з післяумовою. Їхнє застосування.	Лекції, лабораторні заняття, самостійна робота	4
5	Функції мови C/C++. Оголошення масивів. Методи обробки масивів. Перебір елементів. Багатовимірні масиви.	Лекції, лабораторні заняття, самостійна робота	4
6	Шаблони функцій. Поліморфізм функцій, правило простого поліморфізму функцій. Шаблони функцій, їхнє застосування. Inline-функції, особливості їхнього виконання.	Лекції, лабораторні заняття, самостійна робота	4
7	Застосування синонімів і оператора typedef. Простори імен. Функція main.	Лекції, лабораторні заняття, самостійна робота	4

	Застосування оператора typedef для створення синонімів імен і функцій. Спрощення синтаксису при застосуванні покажчиків на функцію. Застосування синонімів byte, word, uint. Застосування синонімів математичних функцій. Поняття простору імен, іменовані та не іменовані простори, що створюються автоматично. Правила створення локальних просторів. Директива using. Створення іменованих просторів імен, особливості застосування. Кваліфікування простору імен.		
8	Функції введення/виведення мови C. Операції читання та запису, а також роботи з файлами в стилі мови C.	Лекції, лабораторні заняття, самостійна робота	4
9	Робота з потоками в стилі мови C++. Створення потоків ifstream, ofstream, зв'язування їх з файлами. Операції введення/виведення. Керування форматами. Безформатне введення/виведення. Файли з довільним доступом.	Лекції, лабораторні заняття, самостійна робота	6
10	Символьні масиви. Рядки в стилі C++. Функції бібліотеки для роботи з рядками.	Лекції, лабораторні заняття, самостійна робота	4

8. Самостійна робота

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.

Основні види самостійної роботи, запропоновані студентам:

- вивчення лекційного матеріалу;
- робота з рекомендованою літературою;
- вивчення окремих тем або питань, що передбачені для самостійного опрацювання;
- вивчення основних термінів та понять з галузі регресійного аналізу;
- підготовка до лабораторних занять;
- підготовка до проміжного та підсумкового контролю;
- контрольна перевірка кожним студентом особистих знань за питаннями для самостійного поглибленого вивчення та самоконтролю;
- робота над індивідуальним завданням.

9. Система та критерії оцінювання курсу

Проміжний контроль знань студентів включає такі заходи:

- виконання та захист лабораторних робіт, що передбачені учбовим планом;
- аудиторні контрольні роботи (2 за семестр), що проводяться перед кожним підсумковим контролем.

Підсумковий контроль знань студентів включає: диф. залік.

Оцінювання знань при проміжному контролі можливо таким чином:

- за відмінний захист лабораторних робіт – 50 балів;
- при відмінному виконанні всіх завдань підсумкової контрольної роботи (контрольної роботи для заочної форми) – 50 балів.

10. Політика курсу

Академічна доброчесність. Студенти не видають за свої результати роботи інших людей. При використанні чужих ідей і тверджень у власних роботах обов'язково посилаються на використані джерела інформації. Під час оцінювання результатів навчання не користуються недозволеними засобами, самостійно виконують навчальні завдання, завдання поточного та підсумкового контролю.

Відвідування занять. Очікується, що всі студенти відвідають усі лекції та практичні заняття курсу. За згоди декана та викладача дозволяється перейти на індивідуальний графік занять. У будь-якому випадку студенти зобов'язані дотримуватися термінів виконання усіх видів робіт, передбачених робочою програмою курсу.

11. Література

1. Бондарев В.М. Программирование на C++. – Харьков, 2005. – 284 с.
2. Шпак З. Програмування мовою C. – Львів : "Львівська політехніка", 2011. – 436 с.
3. Васильєв О. Програмування C++ в прикладах і задачах. – К. : Ліра-К, 2017. – 382 с.
4. Девіс С. C++ для чайників. – К. : Диалектика, 2020. – 800 с.
5. Глушаков С.В., Коваль А.В., Смирнов С.В. Язык программирования C++: Учебный курс. – Харьков: Фолио; М.: Изд. АСТ, 2001. – 500 с.
6. Прата С. C/C++. Язык программирования C++. Лекции и упражнения – К.: Диалектика, 2016. – 1244 с.
7. Б. Страуструп. Язык программирования C++. Краткий курс. – К. : «Вильямс», 2020. – 1000 с.
8. Пінчук В.П., Лозовська Л.І. Програмування мовою C/C++ з прикладами та вправами. Навч. пос. – Запоріжжя: ЗНТУ, 2009. – 204 с.

12. Публікації з тематики курсу

1. Shyrokora, D. V. Evolution of the Ni-Al Janus-like clusters under the impacts of low-energy Ar and Ar13 projectiles / D. V. Shyrokora, G. V. Kornich, S. G. Buga // *Materials Today Communications*. – V. 23. (2020) 101107-12 (Изд. Elsevier) – doi: 10.1016/j.mtcomm.2020.101107.
2. Shyrokora D.V. Formation of the core-shell structures from bimetallic Janus-like nanoclusters under low-energy Ar and Ar13 impacts: A molecular dynamics study / D. V. Shyrokora, G. V. Kornich, S. G. Buga // *Computational Materials Science*. – 2019. – V. 159. – P. 110-119.
3. Ширококорд Д. В. Моделирование столкновительной стадии эволюции двудольных биметаллических кластеров под действием димеров аргона низких энергий / Д. В. Ширококорд, Г. В. Корнич // *Металлофиз. новейшие технол.* – 2017. – Т. 39, № 2. – С. 151–163.