

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Кафедра комп'ютерних наук

ЗАТВЕРДЖУЮ»
Декан факультету інформаційних
технологій
Олена ГЛАЗУНОВА
« 2 серпня » 20 23 р.



«СХВАЛЕНО»
на засіданні кафедри комп'ютерних наук
Протокол № 12 від « 01 » 06 20 23
р.

Завідувач кафедри
Белла ГОЛУБ

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ПРОГРАМУВАННЯ

ЗАГАЛЬНО УНІВЕРСИТЕТСЬКИЙ ВИБІРКОВА

Факультет інформаційних технологій

Розробники: асистент кафедри комп'ютерних наук, к.е.н., Назаренко В.А.

Київ – 2023 р.

Опис навчальної дисципліни

ПРОГРАМУВАННЯ

| Галузь знань, спеціальність, освітня програма, освітній ступінь | |
|--|-----------------------------|
| Освітній ступінь | <i>Бакалавр</i> |
| Характеристика навчальної дисципліни | |
| Вид | Вибіркова |
| Загальна кількість годин | 120 год. |
| Кількість кредитів ECTS | 4 |
| Кількість змістових модулів | 2 |
| Курсовий проект (робота) (за наявності) | - |
| Форма контролю | <i>Залік</i> |
| Показники навчальної дисципліни для денної та заочної форм навчання | |
| | <u>денна форма навчання</u> |
| Курс (рік підготовки) | |
| Семестр | |
| Лекційні заняття | <i>15 год.</i> |
| Лабораторні заняття | <i>15 год.</i> |
| Самостійна робота | <i>90 год.</i> |
| Кількість тижневих аудиторних годин для денної форми навчання | <i>2 год.</i> |

1. Мета, завдання та компетентності навчальної дисципліни

Метою викладання дисципліни «Програмування» є отримання студентами знань з області розробки алгоритмів та програмування. Оволодіння такими знаннями дозволить реалізовувати задачі автоматизації обробки інформації, автоматизації керування об'єктами, в тому числі, сільськогосподарськими, за допомогою комп'ютерної техніки. Такі знання майбутній спеціаліст зможе застосовувати як при подальшому навчанні, так і після отримання вищої освіти у своїй професійній діяльності.

Завдання - Студент повинен знати:

- визначення алгоритмів;
- технології розробки алгоритмів;
- використання алгоритмічних мов для реалізації розроблених алгоритмів на персональних комп'ютерах (ПК);
- технології розробки програм на мові C; – базових конструкцій мови C; оволодіти:
- практичними навичками розробки алгоритмів лінійної, розгалуженої та циклічної структур;
- умінням розроблювати програми на мові C, C++;
- практичними навичками реалізації алгоритмів на мові C, C++ та налагодження програм;
- умінням самостійно опанувати нові методи та технології розробки програм.

Викладання дисципліни ґрунтується на знаннях по таких напрямках як:

- іноземна мова (бажано, англійська);
- вища математика;
- архітектура комп'ютера.

2. Програма та структура навчальної дисципліни для:

– повного терміну денної форми навчання

Модуль №1. Організація програм

Тема №1. Введення в програмування (2 години)

Алгоритми та програми. Мови програмування. Блок-схема як засіб графічного зображення алгоритмів. Лінійні алгоритми. Алгоритми з розгалуженням. Цикли та циклічні структури.

Тема №2. Основні поняття мови C (4 години)

Історія мови C. Характеристика C-систем.

Загальна структура програми на мові C. Директиви передпроцесора. Стандартні функції введення/виведення мови C. Використання коментарів.

Об'ява змінних. Загальні поняття щодо типів даних мови C. Числові типи та символний тип.

Загальні поняття щодо пам'яті комп'ютера. Руйнуюче та не руйнуюче зчитування. Арифметика в мові C. Пріоритет виконання операцій.

Прийняття рішення: операції рівності та відношення. Пріоритет виконання розглянутих операцій. Ключові слова.

Тема №3. Оператори мови C (8 годин)

Операції присвоєння. Повна та скорочена форма операції присвоєння. Операції інкремента та декремента.

Структури управління. Структура вибору *if*. Структура вибору *if/else*.

Структура із множинним вибором *switch*. Приклади програм з реалізованими структурами управління.

Структури повторення. Структура повторення *for*. Блок-схема алгоритму структури *for*. Синтаксис структури *for*. Приклади структур *for*.

Структура повторення *while*. Структура повторення *do/while*. Рекомендації щодо вибору структур повторення в залежності від алгоритму. Приклади програм з реалізованими структурами повторення.

Оператори *break i continue*. Логічні операції. Пріоритет виконання розглянутих операцій.

Тема №4. Проектування програм з використанням покажчиків (8 годин)

Ідеологія використання покажчиків в програмі на мові C. Об'ява та ініціалізація покажчиків. Приклади використання покажчиків.

Операції з покажчиками. Виведення значень покажчиків як адрес оперативної пам'яті. Виведення значення, на яке посилається покажчик. Вирази та арифметика з покажчиками.

Доступ до апаратних елементів комп'ютера за допомогою покажчиків. Приклади програм з використанням покажчиків.

Модуль №2. Масиви і алгоритми обробки

Тема №5. Масиви (8 годин)

Масиви як структурований тип даних. Визначення масивів. Синтаксис об'яви масивів. Поняття “елемент масиву” та “індекс масиву”. Синтаксис використання масивів.

Зв'язок покажчиків з масивами. Вирази та арифметика з покажчиками.

Динамічний розподіл пам'яті.

Стандартні алгоритми роботи з масивами: визначення екстремальних значень, статистичних показників, упорядкування масивів, пошук визначеного елемента.

Багатовимірні масиви. Приклади програм з багатовимірними масивами. Реалізація алгоритмів, пов'язаних з обробкою матриць.

Файли довільного доступу. Створення файлів довільного доступу. Приклади програм.

Довільний запис у файл довільного доступу. Послідовне читання даних із файлів довільного доступу.

| Назви змістових модулів і тем | Кількість годин | | | | | | | |
|-------------------------------|---|--------|--------------|---|---------|-----|------|--|
| | денна форма | | | | | | | |
| | тижні | усього | у тому числі | | | | | |
| | | | л | п | ла б | інд | с.р. | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
| | Семестр I | | | | | | | |
| | Змістовий модуль 1. Організація програм | | | | | | | |

| | | | | | | | |
|---|-----------|-----------|-----------|-----------|---|---|-----------|
| Тема 1. Введення в програмування | 4 | 4 | 2 | 2 | | - | 1 |
| Тема 2. Основні поняття мови C | 4 | 4 | 2 | 2 | | - | 2 |
| Тема 3. Оператори мови C | 4 | 4 | 2 | 2 | | - | 2 |
| Тема 4. Проектування програм з використанням покажчиків | 4 | 4 | 2 | 2 | | | 2 |
| <i>Разом за змістовим модулем 1</i> | <i>16</i> | <i>16</i> | <i>8</i> | <i>8</i> | - | - | <i>7</i> |
| Змістовий модуль 2. Масиви і алгоритми обробки | | | | | | | |
| Тема 5. Масиви | 14 | 14 | 7 | 7 | | - | 4 |
| <i>Разом за змістовим модулем 2</i> | <i>14</i> | <i>14</i> | <i>7</i> | <i>7</i> | | - | <i>4</i> |
| Усього годин | 30 | 30 | 15 | 15 | | | 15 |

3. Теми лабораторних занять

| № з/п | Назва теми |
|-----------|--|
| Семестр I | |
| 1 | C-системи. Змінні та типи змінних, об'ява змінних. Виведення та введення даних. Оператор присвоєння. |
| 2 | Проста та скорочена форми операції присвоєння. Інкремент та декремент. Структура вибору <i>if, if/else</i> . |
| 3 | Структура прийняття рішення switch |
| 4 | Логічні операції |
| 5 | Структура повторення <i>for</i> . |
| 6 | Структура повторення <i>while</i> (повторення з передумовою або повторення, що керуються контрольним значенням). |
| 7 | Структура повторення <i>do/while</i> . |
| 8 | Покажчики. Об'ява та ініціалізація змінної-покажчика. |
| 9 | Масиви. Визначення масивів. Багатовимірні масиви. |

4. Теми самостійної роботи

| № з/п | Назва теми | Кількість годин |
|-------|--|-----------------|
| 1 | Порівняльна характеристика мов програмування | 6 |
| 2 | Методології розробки програм: низхідне та висхідне проектування | 12 |
| 3 | Виведення даних із визначеною шириною поля і точністю представлення. Використання прапорців в рядку управління форматів функції <i>printf</i> . Друк літералів та Esc-послідовностей | 12 |
| 4 | Стандартні бібліотеки компілятора мови | 18 |
| 5 | Алгоритми сортування великих масивів | 30 |

| | | |
|---------------|---|------------|
| 6 | Завершення роботи програми за допомогою макросів <i>exit, atexit</i> | 12 |
| 7 | Основні директиви передпроцесора мови C. Використання директив #include, #define, #ifdef | 12 |
| 8 | Додаткові директиви передпроцесора #error, #pragma . Операції #, ## | 18 |
| 9 | Структури, які посилаються самі на себе. Зв'язані списки, черги, дерева | 30 |
| 10 | Рекурсія – переваги і недоліки | 12 |
| 11 | Низько рівневий доступ до файлів | 18 |
| Всього | | 180 |

5. Зразки контрольних питань, тестів для визначення рівня засвоєння знань студентами.

Комплекти тестів розташовані в осередку ЕНК за посиланням:
<https://elearn.nubip.edu.ua/course/view.php?id=4424>.

6. Методи навчання.

Форми навчання – лекції, лабораторні заняття, самостійна робота, індивідуальна робота – курсовий проект.

При викладанні дисципліни використовуються наступні методи навчання:

- М1. Лекція (проблемна, інтерактивна)
- М2. Лабораторна робота – для використання набутих знань до розв'язування практичних завдань;
- М3. Проблемне навчання – створення проблемної ситуації для зацікавленого і активного сприйняття матеріалу.
- М4. Проектне навчання (індивідуальне, малі групи, групове)

7. Форми контролю.

При викладанні дисципліни передбачені такі форми контролю:

- МК1. Тестування
- МК2. Контрольне завдання
- МК4. Методи усного контролю
- МК5. Екзамен
- МК7. Звіт

Для студентів денної форми навчання: усне опитування (МК4) та експрес контроль (МК1) на лабораторних заняттях, захист індивідуальних лабораторних завдань (МК7), аудиторні модульні контрольні роботи (МК2).

8. **Розподіл балів, які отримують студенти.** Оцінювання знань студента відбувається за 100-бальною шкалою і переводиться в національні оцінки згідно з табл. 1 «Положення про екзамени та заліки у НУБіП України» (наказ про уведення в дію від 26.04.2023 р. протокол № 10)

| Рейтинг студента, бали | Оцінка національна за результати складання | |
|------------------------|--|---------------|
| | екзаменів | заліків |
| 90-100 | Відмінно | Зараховано |
| 74-89 | Добре | |
| 60-73 | Задовільно | |
| 0-59 | Незадовільно | Не зараховано |

Для визначення рейтингу студента (слухача) із засвоєння дисципліни $R_{\text{дис}}$ (до 100 балів) одержаний рейтинг з атестації (до 30 балів) додається до рейтингу студента (слухача) з навчальної роботи $R_{\text{НР}}$ (до 70 балів): $R_{\text{дис}} = R_{\text{НР}} + R_{\text{ат}}$.

| Поточний контроль | | | | Рейтин г з навчальної роботи $R_{\text{НР}}$ | Рейтин г з додаткової роботи $R_{\text{ДР}}$ | Рейтин г штрафний $R_{\text{ШТР}}$ | Підсумкова атестація (екзамен чи залік) | Загальна кількість балів |
|--------------------|--------------------|--------------------|--------------------|--|--|------------------------------------|---|--------------------------|
| Змістовий модуль 1 | Змістовий модуль 2 | Змістовий модуль 3 | Змістовий модуль 4 | | | | | |
| 0-100 | 0-100 | 0-100 | 0-100 | 0-70 | 0-20 | 0-5 | 0-30 | 0-100 |

Рейтинг з додаткової роботи $R_{\text{ДР}}$ додається до $R_{\text{НР}}$ і не може перевищувати 20 балів. Він визначається лектором і надається студентам рішенням кафедри за виконання робіт, які не передбачені навчальним планом, але сприяють підвищенню рівня знань студентів з дисципліни.

Рейтинг штрафний $R_{\text{ШТР}}$ не перевищує 5 балів і віднімається від $R_{\text{НР}}$. Він визначається лектором і вводиться рішенням кафедри для студентів, які матеріал

змістового модуля засвоїли невчасно, не дотримувалися графіка роботи, пропускали заняття тощо.

8. Навчально-методичне забезпечення

1. Електронний навчальний курс, розроблений на базі платформи LMS Moodle, розміщений на навчальному порталі факультету інформаційних технологій за адресою: <https://elearn.nubip.edu.ua/course/view.php?id=4424>

2. ПОЛОЖЕННЯ про екзамени та заліки у Національному університеті біоресурсів і природокористування України, затверджене Вченою радою НУБіП України № 8 від «26» квітня 2023 р.

9. Рекомендовані джерела інформації

Базові

1. Програмування на мові С : навчальний посібник до вивчення дисципліни “Програмування” для студентів зі спеціальностей 121 – «Інженерія програмного забезпечення» 122 – «Комп’ютерні науки» 123 – «Комп’ютерна інженерія» / Б.Л. Голуб, Ю.Є. Бояринова. - К. : , 2017. - 173 с.

2. Програмування на мові С : методичні вказівки до виконання лабораторних робіт з дисципліни «Програмування» для студентів спеціальності: 121 «Інженерія програмного забезпечення», 122 «Комп’ютерні науки», 123 «Комп’ютерна інженерія» / уклад.: Б. Л. Голуб, С. В. Циба, Ю. Є. Бояринова. - К. : , 2016. - 95 с.

3. Вінник В.Ю. Алгоритмічні мови та основи програмування: мова С / В.Ю. Вінник. – Житомир : ЖДТУ, 2007. – 328 с.

4. Ковалюк Т. В. Алгоритмізація та програмування: підручник з грифом МОН України / Т.В. Ковалюк. – Львів : Магнолія-2006, 2013. – 400 с.

Додаткові

5. ANSI, American National Standard for Information Systems – Programming Language C. – New York, 1990.

6. ДСТУ 3008-95. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення.

7. Пол Іре. Об’єктно-орієнтоване програмування з використанням С++: Пер. з англ. - Київ: НІПФ “ДіаСофт Лтд, 1995.

8. Vjarne Stroustrup The C++ Programming language, Addison Weasley, 1986.

9. Фейсон Т. Об’єктно-орієнтоване програмування на Borland C++ 4.5: Пер. з англ. - Київ: Діалектика, 1996. 544с. 5. Сван Т. Опанування Borland C++ 4.5: Пер. з англ. - Київ: Діалектика, 1996. 544с.

10. Object-Oriented Data Structures in C++ (Coursera). URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title0>

11. Beginning C++ Programming Course — Beginner to Beyond (Udemy) URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title1>

12. Coding for Everyone: C and C++ Specialization (Coursera) URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title2>

13. Unreal Engine 4 Mastery: Create Multiplayer Games with C++ (Udemy) URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title3>
14. Learn C++ for Game Development (Udemy) URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title4>
15. C++: від початку до експерта (Udemy) URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title5>
16. Learn Advanced C++ Programming (Udemy) URL: <https://dev.ua/news/valid-article-1676469637-1676469637#title6>