

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ**

Система забезпечення якості освітньої діяльності та вищої освіти

Кафедра комп'ютерних технологій і мехатроніки

ЗАТВЕРДЖЕНО

Гарант освітньо-професійної програми
«Інженерія програмного забезпечення»
першого(бакалаврського) рівня освіти:

зав. каф. КТМ, д.т.н., проф.

 Ніконов О.Я.

СИЛАБУС
АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ /
ALGORITHMIZATION AND PROGRAMMING

| | |
|-------------------------|---|
| освітній ступінь | бакалавр / bachelor |
| галузь знань | 12 Інформаційні технології / Information Technology |
| спеціальність | 121 Інженерія програмного забезпечення / Software Engineering |
| освітня програма | Програмне забезпечення систем / Systems Software |

Харків 2020

Автор: Шуляков Владислав Миколайович, асистент кафедри комп'ютерних технологій і мехатроніки.

Силабус розглянуто та затверджено на засіданні кафедри комп'ютерних технологій і мехатроніки, протокол № 20 від «28» серпня 2020 р.

СИЛАБУС

АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ /

ALGORITHMIZATION AND PROGRAMMING

SYLLABUS

| | |
|-------------------------|--|
| освітній ступінь | бакалавр / bachelor |
| галузь знань | 12 Інформаційні технології / Information Technology |
| спеціальність | 121 Інженерія програмного забезпечення / Software Engineering |
| освітня програма | Програмне забезпечення систем / Systems Software |

Анотація курсу

1. Викладачі

1.1. Лектор: Шуляков Владислав Миколайович

- асистент кафедри комп'ютерних технологій та мехатроніки;
- педагогічний стаж – 13 років;
- контактний телефон +38-093-18-57-595
- e-mail: jasonvlad2013@gmail.com
- наукові інтереси: інтелектуальний аналіз даних, програмування Java, штучні нейронні мережі та нечітка логіка, створення Web сайтів, SEO, HTML/CSS/JS.

1.2. Асистент лектора:

- Шапошнікова Олена Павлівна, доцент, кандидат технічних наук.
- Неронов Сергій Миколайович, старший викладач

2. Дисципліна «Алгоритмізація та програмування»

- рік навчання: 1;
- семестр навчання: 2 ;
- кількість годин за семестр: 240, в т. ч.
лекційних: 32;
практичних занять: 48;
на самостійне опрацювання: 160;
- кількість аудиторних годин на тиждень
лекційних: 2 (раз на тиждень);
практичних занять: 2 (над рискою) / 4 (під рискою).

3. Час та місце проведення

- аудиторні заняття – відповідно до розкладу ХНАДУ, ауд. 214, 313;
- позааудиторна робота – самостійна робота студента із використанням Visual Studio 2019.

4. Мета, завдання та результати вивчення дисципліни (Компетентності), її місце в освітньому процесі

4.1. Метою вивчення дисципліни «Алгоритмізація та програмування» є викладення основних понять алгоритмізації і техніки застосування у програмуванні базових алгоритмічних структур (організація програм) і базових структур даних (організація даних).

4.2. Завданнями дисципліни «Алгоритмізація та програмування» є формування у студентів необхідних знань у галузі сучасних технологій програмування; ознайомлення з технічними, алгоритмічними, програмними і технологічними рішеннями у області програмування; вироблення практичних навичок аналітичного та експериментального дослідження основних методів і засобів, що використовуються в області програмування при рішенні обчислювальних задач.

4.3. Предметом вивчення дисципліни є теорія і практика застосування у програмуванні базових алгоритмічних структур і базових структур даних на основі сучасних технологій розробки програмного забезпечення.

4.4. Результати вивчення дисципліни «Алгоритмізація та програмування».

В результаті вивчення дисципліни студенти повинні:

- аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки;
- знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення;
- проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування;
- знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань;
- застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення;
- знати та вміти застосовувати інформаційні технології обробки,

зберігання та передачі даних;

- вміти проводити розрахунок економічної ефективності програмних систем.

Дисципліна «Алгоритмізація та програмування» базується на знаннях, отриманих студентами під час вивчення таких навчальних дисциплін:

- «Основи інформатики» середньої школи;
- Основи інформаційних технологій;
- Вища математика,

Викладання дисципліни розраховано на виконання студентами лабораторних робіт та самостійних завдань у програмному середовищі Visual Studio.

5. План вивчення дисципліни

| Результати навчання | Навчальна діяльність | Робочий час студента | Оцінювання в балах |
|--|---|----------------------|--------------------|
| Тема 1. Архітектура комп'ютерів, принципи фон Неймана | | | |
| Загальні та спеціальні компетентності: здатність до абстрактного мислення, аналізу та синтезу. Результати навчання: аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки. | Лекція №1 Архітектура комп'ютерів, принципи фон Неймана. Основні питання: 1. Архітектура комп'ютера. 2. Принципи фон Неймана. 3. Етапи розвитку ЕОМ. 4. Функції складових частин ПК. Список рекомендованих джерел: Основний - 5, 6 | 2 | |
| | Задання для самостійної роботи: 1. Архітектура комп'ютера. 2. Принципи фон Неймана. 3. Етапи розвитку ЕОМ. 4. Технічні характеристики складових частин ПК. | 10 | 2 |
| | Практичне заняття 1 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення консольних додатків (Частина 1). План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 2 | 4 |

| Тема 2. Алгоритмізація обчислювальних процесів | | | |
|---|---|-----------|----------|
| Загальні та спеціальні компетентності: здатність вчитися і оволодівати сучасними знаннями. Результати навчання: знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення. | Лекція №2 Алгоритмізація обчислювальних процесів. Основні питання: 1. Етапи розробки обчислювальних задач 2. Способи подання алгоритмів 3. Основні елементи графічного способу 4. Етапи налагодження програм Список рекомендованих джерел: Основний - 6 | 2 | |
| | Задання для самостійної роботи: 1. Етапи розробки обчислювальних задач. 2. Способи подання алгоритмів. 3. Основні елементи графічного способу. 4. Етапи налагодження програм. | 10 | 2 |
| | Практичне заняття 2 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення консольних додатків (Частина 2). План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 4 | 4 |
| Тема 3. Види алгоритмів | | | |
| Загальні та спеціальні компетентності: здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення відповідності вимогами замовника, технічним завданням та стандартами. Результати навчання: проводити передпроектне обстеження предметної | Лекція №3 Види алгоритмів. Основні питання: 1. Лінійні алгоритми. 2. Алгоритми, що розгалужуються. 3. Циклічні алгоритми. 4. Комбіновані алгоритми. Список рекомендованих джерел: Основний - 1,6 | 2 | |
| | Задання для самостійної роботи: 1. Лінійні алгоритми. 2. Алгоритми, що розгалужуються. 3. Циклічні алгоритми. 4. Комбіновані алгоритми | 10 | 2 |
| | Практичне заняття 3 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення консольних додатків. Введення-виведення даних в консольному додатку (Частина 1). План заняття: 1. Актуалізація теоретичного матеріалу. | 2 | 4 |

| | | | |
|---|--|-----------|----------|
| області, системний аналіз об'єкта проектування. | 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | | |
| Тема 4. Основні поняття мови С # | | | |
| Загальні та спеціальні компетентності: здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення. Результати навчання: знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань. | <p style="text-align: center;">Лекція №4</p> <p>Основні поняття мови С #.</p> <p>Основні питання:</p> <ol style="list-style-type: none"> 1. Основні поняття мови С #. 2. Типи даних. 3. Змінні. 4. Перетворення і приведення типів. <p>Список рекомендованих джерел: Основний - 1, 2, 3, 6, 8, 9 Додатковий - 12-17 Інтернет-ресурси - 18</p> | 2 | |
| | <p>Задання для самостійної роботи:</p> <ol style="list-style-type: none"> 1. Структура головного вікна ІСР Visual Studio 2019. 2. Можливості редактора коду Visual Studio. 3. Методика створення консольного застосування у Visual Studio. 4. Створення поліпшеного консольного додатку у Visual Studio. | 10 | 2 |
| | <p style="text-align: center;">Практичне заняття 4</p> <p>Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення консольних додатків. Введення-виведення даних в консольному додатку (Частина 2)</p> <p style="text-align: center;">План заняття:</p> <ol style="list-style-type: none"> 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 4 | 4 |
| Тема 5. Оператори мови С # | | | |
| Загальні та спеціальні компетентності: Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення. | <p style="text-align: center;">Лекція №5</p> <p>Оператори мови С #.</p> <p>Основні питання:</p> <ol style="list-style-type: none"> 1. Оператор присвоювання. 2. Арифметичні оператори. 3. Логічні оператори. 4. Пріоритет операторів. <p>Список рекомендованих джерел: Основний - 1, 2, 3, 6, 8, 9 Додатковий -12-17 Інтернет-ресурси – 18</p> | 2 | |
| | <p>Задання для самостійної роботи:</p> <ol style="list-style-type: none"> 1. Цілочисельні типи. | 10 | 2 |

| | | | |
|---|--|-----------|----------|
| Результати навчання: застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення. | 2. Символьний тип. 3. Речові типи. 4. Логічний тип. | | |
| | <p align="center">Практичне заняття 5</p> Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків лінійної структури з використанням арифметичних операцій. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 2 | 4 |
| Тема 6. Лінійний алгоритм і його реалізація | | | |
| Загальні та спеціальні компетентності: Здатність до абстрактного мислення, аналізу та синтезу. Результати навчання: знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних; вміти проводити розрахунок економічної ефективності програмних систем. | <p align="center">Лекція №6</p> Лінійний алгоритм і його реалізація. Основні питання: 1. Структура програми. 2. Методика розробки програм. 3. Лінійний алгоритм. Список рекомендованих джерел: Основний - 1, 2, 3, 6, 8, 9 Додатковий - 12-17 | 2 | |
| | Задання для самостійної роботи: 1. Структура програми. 2. Методика розробки програм. 3. Лінійний алгоритм. 4. Оператор присвоювання. 5. Арифметичні оператори. 6. Перетворення і приведення типів. | 10 | 2 |
| | <p align="center">Практичне заняття 6</p> Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з структур, що розгалужується. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 4 | 4 |
| Тема 7. Алгоритми, що розгалужуються, і їх реалізація | | | |
| Загальні та спеціальні компетентності: здатність вчитися і оволодівати сучасними знаннями. | <p align="center">Лекція №7</p> Алгоритми, що розгалужуються, і їх реалізація. Основні питання: 1. Базові конструкції структурного програмування. 2. Оператор розгалуження if... else. | 2 | |

| | | | |
|--|--|-----------|----------|
| <p>Результати навчання: аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.</p> | <p>3. Оператор розгалуження switch. 4. Тернарний оператор. Список рекомендованих джерел: Основний - 1, 2, 3, 6, 8, 9 Додатковий -12-17 Інтернет-ресурси - 18</p> | | |
| | <p>Задання для самостійної роботи: 1. Базові конструкції структурного програмування. 2. Оператор розгалуження if... else. 3. Оператор розгалуженій switch. 4. Тернарний оператор.</p> | 10 | 2 |
| | <p>Практичне заняття 7 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків циклічної структури. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.</p> | 2 | 4 |
| Тема 8. Циклічні алгоритми і їх реалізація | | | |
| <p>Загальні та спеціальні компетентності: здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p> <p>Результати навчання: знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.</p> | <p>Лекція №8 Циклічні алгоритми і їх реалізація. Основні питання: 1. Оператор while. 2. Оператор do ... while. 3. Оператор for. 4. Оператор foreach. Список рекомендованих джерел: Основний - 1, 2, 3, 6, 8, 9 Додатковий -12-17 Інтернет-ресурси - 18</p> | 2 | |
| | <p>Задання для самостійної роботи: Види циклічних операторів і особливості їх функціонування.</p> | 10 | 2 |
| | <p>Практичне заняття 8 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків циклічної структури (оператори while, do... while). План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.</p> | 4 | 4 |

| Тема 9. Функції | | | |
|---|---|-----------|----------|
| Загальні та спеціальні компетентності: здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення. Результати навчання: проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування. | Лекція №9 Функції. Основні питання: 1. Призначення функції. 2. Опис і виклик функції. 3. Передача параметрів. 4. Перевантаження функцій. Список рекомендованих джерел: Основний - 2, 3, 6, 8, 9 Додатковий – 12-17 Інтернет-ресурси - 18 | 2 | |
| | Задання для самостійної роботи: 1. Призначення функції. 2. Опис і виклик функції. 3. Передача параметрів. 4. Перевантаження функцій. | 10 | 2 |
| | Практичне заняття 9 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків, що використовують функції (Частина 1). План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 2 | 4 |
| Тема 10. Методологія розробки програм: низхідне та висхідне проектування, модульне програмування | | | |
| Загальні та спеціальні компетентності: здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення. Результати навчання: знати і застосовувати методи розробки алгоритмів, конструювання програмного | Лекція №10 Методологія розробки програм: низхідне та висхідне проектування, модульне програмування. Основні питання: 1. Цілі структурного програмування. 2. Програмування з використанням покрокової деталізації. 3. Низхідне та висхідне програмування. 4. Модульне програмування. Список рекомендованих джерел: Основний – б | 2 | |
| | Задання для самостійної роботи: 1. Цілі структурного програмування. 2. Програмування з використанням покрокової деталізації. 3. Низхідне та висхідне програмування. 4. Модульне програмування. | 10 | 2 |

| | | | |
|---|--|----|---|
| забезпечення та структур даних і знань. | <p align="center">Практичне заняття 10</p> <p>Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків, що використовують функції (Частина 2).</p> <p align="center">План заняття:</p> <ol style="list-style-type: none"> 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 4 | 4 |
| Тема 11. Масиви | | | |
| <p>Загальні та спеціальні компетентності: здатність до абстрактного мислення, аналізу та синтезу.</p> <p>Результати навчання:</p> | <p align="center">Лекція №11</p> <p>Масиви. Основні питання: 1. Одномірні масиви. 2. Ініціалізація масивів. 3. Багатовимірні масиви. 4. “Рвані” масиви. Список рекомендованих джерел: Основний - 2, 3, 6, 8, 9 Додатковий - 12-17 Інтернет-ресурси - 18</p> | 2 | |
| застосовувати на практиці інструментальні програмні засоби | <p>Задання для самостійної роботи:</p> <ol style="list-style-type: none"> 1. Оголошення одновимірних масивів. 2. Оператори, що використовуються для обробки масивів. | 10 | 2 |
| доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення. | <p align="center">Практичне заняття 11</p> <p>Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків по обробці одновимірних масивів даних.</p> <p align="center">План заняття:</p> <ol style="list-style-type: none"> 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 2 | 4 |
| Тема 12. Рядки | | | |
| <p>Загальні та спеціальні компетентності: здатність вчитися і оволодівати сучасними знаннями.</p> | <p align="center">Лекція №12</p> <p>Рядки. Основні питання: 1. Створення рядків. 2. Робота з рядками. 3. Масиви рядків. 4. Використання рядків в switch-операторах. Список рекомендованих джерел:</p> | 2 | |

| | | | |
|---|---|-----------|----------|
| Результати навчання: знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних; вміти проводити розрахунок економічної ефективності програмних систем. | Основний - 2, 3, 6, 8, 9 Додатковий -12-17 Інтернет-ресурси - 18 | | |
| | Задання для самостійної роботи: 1. Створення рядків. 2. Функції, що використовуються при обробки рядків. 3. Масиви рядків. 4. Використання рядків в switch-операторах. | 10 | 2 |
| | Практичне заняття 12 Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків по обробці рядків. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 4 | 4 |
| Тема 13. Показчики | | | |
| Загальні та спеціальні компетентності: здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами. | Лекція №13 | | |
| | Показчики. Основні питання: 1. Оголошення показчиків. 2. Оператори “*” і “&” 3. Арифметичні операції над показчиками. 4. Показчики та масиви. Список рекомендованих джерел: Основний - 6, 8, 9 Додатковий -12-17 | 2 | |
| | Задання для самостійної роботи: 1. Оголошення показчиків. 2. Оператори “*” і “&” 3. Арифметичні операції над показчиками. 4. Показчики та масиви. | 10 | 2 |
| Результати навчання: аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням | Практичне заняття 13 Дослідження можливостей інтегрованою середовища розробки Visual Studio для створення додатків, що використовують показчики. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). | 2 | 5 |

| | | | |
|---|--|-----------|----------|
| сучасних досягнень науки і техніки. | 3. Презентація і захист виконаної роботи. 4. Тестування знань. | | |
| Тема 14. Виняткові ситуації | | | |
| Загальні та спеціальні компетентності: здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення. Результати навчання: знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення. | Лекція №14 | 2 | |
| | Виняткові ситуації. Основні питання: 1. Поняття про виняткову ситуацію. 2. Використання try- і catch-блоків. 3. Генерування виключень. 4. Використання блоку finally. Список рекомендованих джерел: Основний - 6, 8, 9 Додатковий -12-17 Інтернет-ресурси -18 | | |
| | Задання для самостійної роботи: 1. Поняття про виняткову ситуацію. 2. Використання try- і catch-блоків. 3. Генерування виключень. 4. Використання блоку finally. | 10 | 2 |
| | Практичне заняття 14 | 4 | 5 |
| | Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків по обробці багатовимірних масивів даних. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | | |
| Тема 15. Файловий ввід і вивід. Текстові файли | | | |
| Загальні та спеціальні компетентності: здатність обгрунтовано обирати та освоювати інструментарій розробки супроводження програмного забезпечення. | Лекція №15 | 2 | |
| | Файловий ввід і вивід. Текстові файли. Основні питання: 1. Відкриття та закриття файлів. 2. Запис байтів в файл. 3. Зчитування байтів з файлів. 4. Зчитування і запис символів з файлів. Список рекомендованих джерел: Основний - 6, 8, 9 Додатковий -12-17 Інтернет-ресурси - 18 | | |
| Результати навчання: | Задання для самостійної роботи: 1. Відкриття та закриття файлів. 2. Запис байтів в файл. 3. Зчитування байтів з файлів. | 10 | 2 |

| | | | |
|---|---|-------------------------------------|------------------|
| проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування. | 4. Зчитування і запис символів з файлів. | | |
| | <p align="center">Практичне заняття 15</p> <p>Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків по обробці текстових файлів (Частина 1).</p> <p align="center">План заняття:</p> <ol style="list-style-type: none"> 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 2 | 5 |
| Тема 16. Файловий ввід і вивід. Бінарні файли | | | |
| <p>Загальні та спеціальні компетентності: здатність до абстрактного мислення, аналізу та синтезу.</p> <p>Результати навчання: знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> | Лекція №16 Файловий ввід і вивід. Бінарні файли. Основні питання: <ol style="list-style-type: none"> 1. Відкриття та закриття файлів. 2. Запис двійкових даних в файл. 3. Зчитування двійкових даних з файлу. 4. Довільний доступ до даних у файлу. Список рекомендованих джерел: Основний - 6, 8, 9 Додатковий -12-17 Інтернет-ресурси - 18 | 2 | |
| | Задання для самостійної роботи: <ol style="list-style-type: none"> 1. Відкриття та закриття файлів. 2. Запис двійкових даних в файл. 3. Зчитування двійкових даних з файлу. 4. Довільний доступ до даних у файлу. | 10 | 2 |
| | <p align="center">Практичне заняття 16</p> <p>Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків по обробці текстових файлів (Частина 2).</p> <p align="center">План заняття:</p> <ol style="list-style-type: none"> 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань. | 4 | 5 |
| | Разом | 240годин/ 8 кредитів | 100 балів |
| Підсумковий контроль | | Екзамен | |

6. Список рекомендованих джерел

Основний

1. Государственный стандарт «Схемы алгоритмов, программ, данных и систем». ГОСТ 19.701-90. - М.: Госстандарт, 1991 -37 с.
2. Ватсон Б. С# 4.0 на примерах. - СПб.: БХВ-Петербург, 2011. -608с.: ил.
3. Петцольд Ч. Программирование для Microsoft Windows на С#. В 2-х томах. Том 1. Пер. с англ. - М.: «Русская Редакция», 2002.- 576 с.: ил.
4. Петцольд Ч. Программирование для Microsoft Windows на С#. В 2-х томах. Том 2. Пер. с англ. - М.: «Русская Редакция», 2002.- 624 с.: ил.
5. Таненбаум Э. Архитектура компьютера. 5-е изд. - СПб.: БХВ-Петербург, 2010. - 844с.: ил.
6. Тімонін В.О. Конспект лекцій з дисципліни «Алгоритмізація та програмування» для студентів за спеціальністю 121 «Інженерія програмного забезпечення». - Харків: ХНАДУ, 2018. - 128 с. (в електронній формі).
7. Тімонін В.О. Методичні вказівки до проведення практичних робіт з дисципліни «Алгоритмізація та програмування» для студентів за спеціальністю 121 «Інженерія програмного забезпечення». - Харків: ХНАДУ, 2018. - 84 с.
8. Троелсен Э. Язык программирования С# 2010 и платформа .NET 4.0. Пер. с англ. - М.: Издательский дом "Вильямс", 2011. — 1392 с.: ил.
9. Шилдт Г. С# 4.0: полное руководство. - М.: ООО «Вильямс», 2012. - 1056 с.: ил.
10. Фленов М.Е. Библия С#. - СПб.: БХВ-Петербург, 2011. - 560с.: ил.

Додатковий

11. Берковський В.В., Левтеров А.І., Костикова М.В., Онуфрей Ю.Є., Подоляка О.О., Попеленко А. А. Програмування в середовищі С(С++). Збірник задач. - Харків: ХНАДУ, 2006. - 224 с.
12. Голощапов А.Л. Microsoft Visual Studio 2010. - СПб.:БХВ-Петербург, 2011. -544 с.: ил.
13. Дейтел Х., Дейтел П. Как программировать в С++. 5-е изд. - СПб.: Бином-Пресс, 2008. — 1113 с.: ил.
14. Вступ до програмування мовою С++. Організація обчислень : навч. посіб. / Ю. А. Белов, Т. О. Карнаух, Ю. В. Коваль, А. Б. Ставровський. - К. : Видавничо-поліграфічний центр "Київський університет", 2012. - 175 с. с.: іл. ISBN (укр.)
15. Лафоре Р. Объектно-ориентированное программирование в С++. Классика Computer Science. 4-е изд. - СПб.: Питер, 2008. - 928с. : ил.
16. Прата С. Язык программирования С++. Лекции и упражнения. 5-е изд.: Пер. с англ. - М.: ООО «Вильямс», 2007. - 1184с.: ил.
17. Страуструп Б. Язык программирования С++. Специальное издание. СПб.: Бином-Пресс, 2008. - 1104 с.: ил.

Інтернет-ресурси

18. Дистанційний курс-ресурс «Алгоритмізація та програмування» [Електронний ресурс]. – Режим доступу: <http://dl.khadi.kharkov.ua/course/view.php?id=1470>.