

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра програмної інженерії та інформаційних технологій управління
(назва)

«ЗАТВЕРДЖУЮ»

Голова науково-методичної комісії _____
(назва комісії)

_____ (підпис) _____ (ініціали та прізвище)

« _____ » _____ 20 _____ року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

АЛГОРИТМИ І СТРУКТУРИ ДАНИХ

(назва навчальної дисципліни)

рівень вищої освіти перший (бакалаврський)
перший (бакалаврський) / другий (магістерський)

галузь знань 12 Інформаційні технології
(шифр і назва)

спеціальність 126 Інформаційні системи та технології
(шифр і назва)

спеціалізація _____
(шифр і назва)

вид дисципліни професійна підготовка
(загальна підготовка / професійна підготовка)

форма навчання денна
(денна / заочна)

Харків – 2017 рік

ЛИСТ ЗАТВЕРДЖЕННЯ

Робоча програма з навчальної дисципліни

АЛГОРИТМИ І СТРУКТУРИ ДАНИХ

(назва дисципліни)

Розробники:

доцент, к.т.н., доц.
(посада, науковий ступінь та вчене звання)

(підпис)

Стратієнко Н.К.
(ініціали та прізвище)

(посада, науковий ступінь та вчене звання)

(підпис)

(ініціали та прізвище)

Робоча програма розглянута та затверджена на засіданні кафедри

(назва кафедри)

Протокол від « 31 » _____ 2017 року № 1

Завідувач кафедри _____
(назва кафедри) (підпис) (ініціали та прізвище)

ЛИСТ ПЕРЕЗАТВЕРДЖЕННЯ РОБОЧОЇ НАВЧАЛЬНОЇ ПРОГРАМИ

Дата засідання кафедри – розробника РПНД	Номер протоколу	Підпис завідувача кафедри	Підпис голови НМК (для дисциплін загальної підготовки та дисциплін професійної підготовки за спеціальністю) або завідувача випускової кафедри (для дисциплін професійної підготовки зі спеціалізації, якщо РПНД розроблена не випусковою кафедрою)

МЕТА, КОМПЕТЕНТНОСТІ, РЕЗУЛЬТАТИ НАВЧАННЯ ТА СТРУКТУРНО-ЛОГІЧНА СХЕМА ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета навчальної дисципліни "Алгоритми і структури даних" - формування у студентів системи знань про базові структури даних і основні обчислювальні алгоритми, а також придбання практичних навичок з проектування, розроблення та аналізу алгоритмів, оцінювання їх ефективності та складності.

Компетентності:

Загальні компетентності:

- Здатність застосовувати знання у практичних ситуаціях.
- Знання та розуміння предметної області та розуміння професійної діяльності.
- Здатність вчитися і оволодівати сучасними знаннями.
- Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

Спеціальні (фахові) компетентності:

Здатність до алгоритмічного та логічного мислення. Здатність до побудови логічних висновків, використання моделей алгоритмічних обчислень, проектування, розроблення та аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем

Нормативний зміст підготовки здобувачів вищої освіти, сформульований у термінах результатів навчання

Класифікація компетентностей за НРК	Знання	Уміння	Комунікація	Автономія та відповідальність
1. Здатність до алгоритмічного та логічного мислення. Здатність до побудови логічних висновків, використання моделей алгоритмічних обчислень, проектування, розроблення та аналізу алгоритмів, оцінювання їх ефективності та складності,	Знання основних алгоритмів та структур даних. Знання базових понять теорії алгоритмів, питань обчислюваності, розв'язності та нерозв'язності масових проблем, понять часової та просторової складності алгоритмів при	Використовувати основні алгоритми та структури даних, встановлювати розв'язність та нерозв'язність алгоритмічних проблем, розробляти та аналізувати алгоритми, оцінювати ефективність складності.	Здатність спілкуватися з колегами, клієнтами, партнерами щодо конкретних питань проектування та моделювання інформаційних і програмних систем, складати їх аналітичні звіти, доповіді у письмовій формі та виступати з	Здатність обґрунтовувати власну точку зору щодо проектування, розроблення та аналізу алгоритмів та використання структур даних при моделюванні предметних областей

розв'язності нерозв'язності алгоритмічних проблем адекватного моделювання предметних областей і створення програмних інформаційних систем	та розв'язанні обчислювальних задач. для		результатами власної роботи на нарадах, конференціях тощо .	
---	--	--	---	--

Структурно-логічна схема вивчення навчальної дисципліни

Попередні дисципліни:	Наступні дисципліни:
Математичний аналіз (частина 1)	Дискретна математика (частина 2)
Основи програмування (частина 1)".	

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

(розподіл навчального часу за семестрами та видами навчальних занять)

Семестр	Загальний обсяг (годин) / кредитів ECTS	з них		За видами аудиторних занять (годин)			Індивідуальні завдання студентів (КП, КР, РГ, Р, РЕ)	Поточний контроль	Семестровий контроль	
		Аудиторні заняття (годин)	Самостійна робота (годин)	Лекції	Лабораторні заняття	Практичні заняття, семінари			Контрольні роботи (кількість робіт)	Залік
1	2	3	4	5	6	7	8	9	10	11
2	120/4	64	56	32	32			2		+
1	90/3	64	26	32	32			2		-

Попередні дані

Поточні дані

Співвідношення кількості годин аудиторних занять до загального обсягу складає 53,3 % (%):

СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№ з/п.	Види навчальних занять (Л, ЛЗ, ПЗ, СР)	Кількість годин	Номер семестру (якщо дисципліна викладається у декількох семестрах). Назви змістових модулів. Найменування тем та питань кожного заняття. Завдання на самостійну роботу.	Рекомендована література (базова, допоміжна)
1	2	3	4	5
Змістовий модуль 1. Базові структури даних та основні обчислювальні алгоритми				
	Л ЛР СР	2 0 4	<p>Тема 1 Вступ до структур даних та алгоритмів Поняття структур даних та їх класифікація.</p> <p>Тема 2 Базові структури даних Формалізація поняття алгоритму. Основні напрямки в теорії алгоритмів. Практичне застосування результатів теорії алгоритмів</p>	[1–3], [7-9]
	Л ЛР СР	4 8 6	<p>Тема 3 Алгоритми сортування, злиття та пошуку Масиви. Стеки. Черги. Зв'язні списки. Хеш-таблиці. Пряма адресація. Хеш-функції. Двійкові дерева пошуку. Червоно-чорні дерева.</p>	[1–6], [7-10]
	Л ЛР СР	2 4 6	<p>Тема 4 Комбінаторні алгоритми Сортування за квадратичний час. Сортування вибором. Сортування обміном. Сортування за $O(n \log n)$. Швидке сортування. Вибір елемента для розділення. Сортування злиттям. Витрати пам'яті при сортуванні злиттям. Злиття послідовностей. Бінарний пошук. Нижні оцінки швидкості сортування. Дерево розв'язків. Сортування підрахунком.</p>	[1–4], [9-11]
	Л ЛР СР	2 2 4	<p>Тема 5 Фундаментальні алгоритми на графах і деревах Підрахунок кількості. Перестановки. Розміщення. Комбінації. Розбиття. Імовірність. Імовірнісний простір. Елементарні події. Неможлива подія. Несумісні події. Аксиоми імовірності. Дискретний розподіл імовірності. Генератори псевдовипадкових чисел. Властивості випадкових та псевдовипадкових чисел. Поширені недоліки генераторів псевдовипадкових чисел. Лінійний конгруентний метод. Вихор Мерсенна.</p>	[1–5], [8-10]
	Л ЛР СР	2 4 4	<p>Тема 5 Фундаментальні алгоритми на графах і деревах Представлення графів. Вершини. Ребра. Орієнтовані та неорієнтовані графи. Список суміжних вершин. Матриця суміжності. Розріджений граф. Зважений граф. Пошук в глибину. Пошук в ширину. Дерева пошуку. Рекурсивна та нерекурсивна реалізації пошуку у графі. Топологічне сортування.</p>	[1–6], [7-11]

	Л ЛР СР	2 2 4	Тема 6 Геометричні алгоритми Властивості відрізків. Опукла комбінація. Векторний добуток. Напрямок повороту. Перевірка перехрещення відрізків. Обмежуючий прямокутник. Метод прямої, що рухається. Відношення порядку на відрізках. Побудова опуклої оболонки. Алгоритм Грехема. Алгоритм Джарвіса. Метод додавання точок. Складність алгоритмів побудови опуклої оболонки.	[1–3], [9-11]
	Л ЛР СР	2 0 4	Тема 7 Криптографічні алгоритми Класифікація криптографічних алгоритмів. Криптосистеми із закритим ключем. Поняття закритого ключа. Шифр Цезаря. Шифр Віжинера. Зламування шифрів Цезаря та Віжинера. Криптосистеми з відкритим ключем. Поняття відкритого ключа. Китайська теорема про остачі. Функція Ейлера. Криптографічні хеш-функції. Поняття хеш-функції. Необхідність криптографічного шифрування.	[1–6], [9-11]
	Л ЛР СР	4 0 4	Тема 8 Евристичні алгоритми Оцінка якості наближеного алгоритму. Схема наближення для заданої оптимізаційної задачі. Задача про покриття вершин. Наближений алгоритм для пошуку покриття вершин. Максимальна помилка наближеного алгоритму для пошуку покриття вершин. Задача комівояжера. Нерівність трикутника. Наближений алгоритм для задачі комівояжера. Максимальна помилка наближеного алгоритму для задачі комівояжера.	[1–4], [8-10]
Змістовий модуль 2. Аналіз алгоритмів				
	Л ЛР СР	2 4 4	Тема 9 Математичні основи аналізу алгоритмів Асимптотичні позначення. Швидкість росту функцій. Логарифмічний ріст. Лінійний ріст. Квадратичний ріст. Експоненційний ріст. Стандартні функції та позначення. Суми та їхні властивості. Прогресії. Суми різниць. Оцінки сум. Індукція. Почленне порівняння.	[1–6], [8]
	Л ЛР СР	2 2 4	Тема 10 Рекурсія Алгоритмічна система на основі рекурсивних функцій. Метод підстановки. Способи вгадати оцінку (аналогія, послідовні наближення). Індукція та парадокс винахідника. Заміна змінних. Перетворення на суми. Ітерації співвідношень. Дерево рекурсії. Загальне рішення великого класу рекурентних співвідношень. Основна теорема про рекурсивні оцінки. Приклади використання основної теореми про рекурсивні оцінки.	[1–3], [9]
	Л ЛР СР	2 4 4	Тема 11 Алгоритмічні стратегії Принцип «Розділай і пануй». Розподіл на підзадачі. Динамічне програмування. Задача про множення матриць. Оцінка складності алгоритму розв'язання задачі про множення матриць. Задача про пошук найбільшої спільної підпослідовності. Довжина найбільшої спільної підпослідовності. Відтворення	[1–6], [7-11]

			найбільшої спільної підпоследовності. Жадібні алгоритми. Задача про розподіл заявок. Амортизаційний аналіз. Метод групувань. Метод передплати. Метод потенціалів.	
	Л ЛР СР	2 2 4	Тема 12 Основи теорії обчислюваності Поняття обчислюваності й обчислювальні процедури. Поняття відносного алгоритму й відносної обчислюваності, поняття зведення. Машина Тюринга. Складові частини машини Тюринга. Можливості машини Тюринга. Ілюстрація роботи машини Тюринга для простих алгоритмів. Основна гіпотеза теорії алгоритмів. Алгоритмічно нерозв'язні проблеми.	[1–5], [8-10]
	Л ЛР СР	4 0 4	Тема 13 Класи складності P й NP Поліноміальний час. Ефективний алгоритм. Абстрактна задача. Поліноміальна задача. Формальні мови для задач розв'язуваності. Перевірка належності до мови та клас NP. Задача про гамільтонов цикл у графі. Алгоритм перевірки. NP-важкі й NP-повні задачі. Клас NP.	[1–6], [7-11]
Разом (годин)		120		

САМОСТІЙНА РОБОТА

№ з/п	Назва видів самостійної роботи	Кількість годин
1	Опрацювання лекційного матеріалу	20
2	Підготовка до лабораторних занять	10
3	Самостійне вивчення тем та питань, які не викладаються на лекційних заняттях	20
4	Інші види самостійної роботи	6
	Разом	56

ІНДИВІДУАЛЬНІ ЗАВДАННЯ

Розрахункові завдання

(вид індивідуального завдання)

Завдання 1. Визначити складність алгоритмів, записаних у вигляді псевдокоду:

```
a) for k := 1 to n do
    for i := 1 to n do
        for j := 1 to n do
```

if $a[i, k] + a[k, j] < a[i, j]$ then
 $a[i, j] := a[i, k] + a[k, j]$;

б) $d := b * b - 4 * a * c$;
 $s_d := \text{sqrt}(d)$;
 $x1 := (-b - s_d) / (2 * a)$;
 $x2 := (-b + s_d) / (2 * a)$.

Порівняти попарно швидкості росту заданих функцій, визначивши, чи можна дану пару записати, як $f(n) = O(g(n))$, при цьому $n, k > 1$:

$$f_1(n) = \log^k n,$$

$$f_2(n) = n^k.$$

Довести дані твердження, користуючись визначенням O -позначення:

$$n^2 + 2n = O(n^2),$$

$$\log n = O(n).$$

Завдання 2. Визначити найбільшу спільну підпоследовність:

1) *addaadfbcc* та *cdabdfadcc*;

2) *daaeefeefad* та *deesaaeefa*.

Завдання 3. Розв'язати задачу про заявки жадібним алгоритмом.

1) [12,15); [1,3); [14,16); [4,5); [10,13); [7,13); [8,10); [5,6); [7,10); [5,8);

2) [5,6); [10,16); [14,18); [10,12); [13,16); [1,5); [9,11); [7,12); [7,8); [9,12).

Завдання 4. Написати програму для машини Тьюринга, яка додає два числа. Перевірити її роботу на числах a, b . Для визначення чисел необхідно додати 10 до номера варіанту та взяти першу цифру суми в якості a , другу – як b .

Завдання 5. Виконати сортування послідовностей:

1) 5 9 1 7 1 7 4 8 2 4;

2) 9 6 2 0 5 0 8 7 4 5.

Методи сортування:

1) бульбашкою;

2) кучею;

3) злиттям;

4) швидким сортуванням.

Завдання 6. Згенерувати дві псевдовипадкові послідовності довжиною 20 двома генераторами (початкові параметри вибрати довільно).

Завдання 7. Побудувати орієнтований граф без циклів з 17-ма вершинами та 24-ма ребрами, не більше 5-ти ребер при кожній вершині. Виконати пошук в глибину, ширину та провести топологічне сортування.

Завдання 8. Взяти будь-яку послідовність із завдання 6 та використати її числа, як координати $x_1, y_1; x_2, y_2; \dots x_{10}, y_{10}$. Знайти опуклу оболонку цієї множини точок.

Література: основна [1–6]; додаткова [7-11].

МЕТОДИ НАВЧАННЯ

При викладанні навчальної дисципліни для активізації навчального процесу передбачено застосування сучасних навчальних технологій, таких, як: проблемні лекції; робота в малих групах; семінари-дискусії; кейс-метод; ділові ігри.

Проблемні лекції спрямовані на розвиток логічного мислення студентів. Коло питань теми лекції обмежується двома-трьома ключовими моментами, увага студентів концентрується на матеріалі, що не знайшов широкого відображення в підручниках, використовується досвід закордонних навчальних закладів з роздаванням студентам під час лекцій друкованого матеріалу та виділенням головних висновків з питань, що розглядаються. При викладанні лекційного матеріалу студентам пропонуються питання для самостійного розмірковування. При цьому лектор задає запитання, які спонукають студента шукати розв'язання проблемної ситуації. Така система примушує студентів сконцентруватися і почати активно мислити в пошуках правильної відповіді.

На початку проведення проблемної лекції необхідно чітко сформулювати проблему, яку необхідно вирішити студентам. При викладанні лекційного матеріалу слід уникати прямої відповіді на поставлені запитання, а висвітлювати лекційний матеріал таким чином, щоб отриману інформацію студент міг використовувати при розв'язанні проблеми.

Міні-лекції передбачають викладання навчального матеріалу за короткий проміжок часу й характеризуються значною ємністю, складністю логічних побудов, образів, доказів та узагальнень. Міні-лекції проводяться, як правило, як частина заняття-дослідження. На початку проведення міні-лекції за вказаними темами лектор акцентує увагу студентів на необхідності представити викладений лекційний матеріал у так званому структурно-логічному вигляді. На розгляд виносяться питання, які зафіксовані у плані лекцій, але викладаються вони стисло. Лекційне заняття, проведене у такий спосіб, пробуджує у студента активність та увагу при сприйнятті матеріалу, а також спрямовує його на використання системного підходу при відтворенні інформації, яку він одержав від викладача. Проблемні лекції та міні-лекції доцільно поєднувати з такою формою активізації навчального процесу, як робота в малих групах.

Робота в малих групах дає змогу структурувати лекційні або лабораторні заняття за формою і змістом, створює можливості для участі кожного студента в роботі за темою заняття, забезпечує формування особистісних якостей та досвіду соціального спілкування. Після висвітлення проблеми (при використанні проблемних лекцій) або стислого викладання матеріалу (при використанні міні-лекцій) студентам пропонується об'єднуватися у групи по 5-6 осіб та презентувати наприкінці заняття своє бачення та сприйняття матеріалу.

Презентації – виступи перед аудиторією, що використовуються для представлення певних досягнень, результатів роботи групи, звіту про виконання індивідуальних завдань. Однією з позитивних рис презентації та її переваг при використанні в навчальному процесі є обмін досвідом, який здобули студенти при роботі у певній малій групі.

Лабораторні заняття (з елементами семінарської дискусії) дозволяють формувати у студентів навички особистого експериментального дослідження фізичних процесів що відбуваються під час роботи компонентів операційної системи, проводити аналіз умов її функціонування, а також розробляти нові елементи та системні компоненти відповідно до вимог, що пред'являються до них, узагальнювати отримані результати, формулювати висновки та думки, вести подальший обмін думками та поглядами з іншими учасниками щодо отриманих результатів досліджень з даної теми, а також розвивають творче мислення, допомагають формувати погляди і переконання, вчать об'єктивно оцінювати результати і пропозиції опонентів, критично підходити до власних результатів та поглядів.

Ділові та рольові ігри – форма активізації студентів, за якої вони задіяні в процесі інсценізації певної виробничої ситуації у ролі безпосередніх учасників подій. Наприклад, при проведенні лабораторного заняття за темою "Дослідження організації пам'яті в захищеному режимі" слід поділити аудиторію на групи, кожній з яких дати завдання використовуючи різноманітні алгоритми розподілу пам'яті визначити ступінь ефективності використання фізичної пам'яті комп'ютера.

Кейс-метод – метод аналізу конкретних ситуацій, який дає змогу наблизити процес навчання до реальної практичної діяльності спеціалістів і передбачає розгляд виробничих, управлінських та інших ситуацій, складних конфліктних випадків, проблемних ситуацій, інцидентів у процесі вивчення навчального матеріалу.

Розподіл форм та методів активізації процесу навчання за темами навчальної дисципліни наведено у табл.

Тема	Практичне застосування навчальних технологій
Тема 2. Базові структури даних	Міні-лекція "Базові структури даних", презентація
Тема 3 Алгоритми сортування, злиття та пошуку	Проблемні лекції з питань „Сортування вибором”, „Сортування обміном”, „Швидке сортування”, „Сортування злиттям”, робота в малих групах, презентація

Тема 5. Фундаментальні алгоритми на графах і деревах	Міні-лекція з питань „Пошук в глибину” „Пошук в ширину”, робота в малих групах, презентація
Тема 6. Геометричні алгоритми	Проблемна лекція з питань „Побудова опуклої оболонки. Алгоритм Джарвісу”, презентація, робота в малих групах
Тема 11. Алгоритмічні стратегії	Міні-лекція з питань „Задача о маршруті на прямокутному полі”, „Задача про пошук найбільшої спільної підпоследовності”, „Задача про розподіл заявок”, робота в малих групах, презентація результатів роботи в малих групах

МЕТОДИ КОНТРОЛЮ

Система оцінювання знань, вмінь та навичок студентів передбачає виставлення оцінок за усіма формами проведення занять. Перевірка та оцінювання знань студентів може проводитись у таких формах:

1. Оцінювання роботи студентів у процесі лабораторних занять.
2. Проведення проміжного контролю.
3. Проведення модульного контролю.

Загальна модульна оцінка складається з поточної оцінки, яку студент отримує під час лабораторних занять та оцінки за виконання модульної контрольної роботи.

Загальна оцінка з дисципліни визначається як середнє арифметичне модульних оцінок.

Порядок поточного оцінювання знань студентів

Поточне оцінювання здійснюється під час проведення практичних занять і має на меті перевірку рівня підготовленості студента до виконання конкретної практичної роботи. Об'єктами поточного контролю є:

- 1) активність та результативність роботи студента протягом семестру над вивченням програмного матеріалу дисципліни; відвідування занять;
- 2) виконання проміжного контролю;
- 3) виконання модульного контрольного завдання.

Контроль систематичного виконання самостійної роботи та активності на практичних заняттях

Оцінювання проводиться за 5-бальною шкалою за такими критеріями:

- 1) розуміння, ступінь засвоєння теорії та методології проблем, що розглядаються;
- 2) ступінь засвоєння матеріалу дисципліни;
- 3) ознайомлення з рекомендованою літературою, а також із сучасною літературою з питань, що розглядаються;
- 4) уміння поєднувати теорію з практикою при розгляді виробничих ситуацій, розв'язанні задач, проведенні розрахунків при виконанні завдань, винесених для самостійного опрацювання, та завдань, винесених на розгляд в аудиторії;
- 5) логіка, структура, стиль викладу матеріалу в письмових роботах і при виступах в аудиторії, вміння обґрунтовувати свою позицію, здійснювати узагальнення інформації та робити висновки.

Оцінка "відмінно" ставиться за умови відповідності виконаного завдання студента або його усної відповіді до всіх п'яти зазначених критеріїв.

Відсутність тієї чи іншої складової знижує оцінку на відповідну кількість балів.

При оцінюванні практичних завдань увага приділяється також їх якості та самостійності, своєчасності здачі виконаних завдань викладачу (згідно з графіком навчального процесу). Якщо якась із вимог не буде виконана, то оцінка буде знижена.

Проміжний модульний контроль

Проміжний модульний контроль рівня знань передбачає виявлення опанування студентом матеріалу лекційного модуля та вміння застосовувати його для вирішення практичної ситуації і проводиться у вигляді тестування. При цьому тестове завдання може містити як запитання, що стосуються суто теоретичного матеріалу, так і запитання, спрямовані на вирішення невеличкого практичного завдання.

Тестове завдання містить запитання одиничного і множинного вибору різного рівня складності. Для оцінювання рівня відповідей студентів на тестові завдання використовуються такі критерії оцінювання:

- оцінка "відмінно" (12 – 10 балів) – виставляється у випадку, якщо студент правильно відповів на 24 – 20 тестових запитань;
- оцінка "дуже добре" (9 балів) – 19 – 18 правильних відповідей;
- оцінка "добре" (8 – 7 балів) – 17 – 14 правильних відповідей;
- оцінка "задовільно" (6 балів) – 13 – 12 правильних відповідей;
- оцінка "достатньо" (5 – 4 балів) – 11 – 8 правильних відповідей;
- оцінка "незадовільно" (3 бали) – 7 – 6 правильних відповідей;
- оцінка "незадовільно" (2 – 1 бали) – 5 – 0 правильних відповідей.

Тести для проміжного контролю обираються із загального переліку тестів за відповідними модулями.

Проведення модульного контролю

Модульний контроль здійснюється та оцінюється за двома складовими: лекційний (теоретичний) модуль та практичний модуль.

Теоретичний модульний контроль проводиться у письмовій формі після того як розглянуто весь теоретичний матеріал. Після вивчення тем 1 – 8 (модуль 1) студенти денної форми навчання виконують – завдання до модуля 1. Відповідно, після вивчення тем 9 – 13 (модуль 2) – завдання до модуля 2.

Практичний модульний контроль проводить після виконаних практичних завдань у межах кожного з двох модулів.

Теоретичне модульне завдання оцінюється за 12-бальною системою відповідно до кваліфікаційних вимог до бакалаврів спеціальності 121 „Інженерія програмного забезпечення”. При цьому вважається, що для набору 1 бала оцінки потрібно правильно відповісти на 2 запитання завдання до модуля.

Загальна оцінка за практичне модульне завдання визначається як середнє арифметичне з оцінок за виконані практичної роботи при округленні в сторону студента.

Підсумкова оцінка з дисципліни розраховується як середня з кількох складових, що враховує оцінки кожного виду контролю (дві оцінки за результатами поточного модульного контролю за роботу протягом семестру).

РОЗПОДІЛ БАЛІВ, ЯКІ ОТРИМУЮТЬ СТУДЕНТИ, ТА ШКАЛА ОЦІНЮВАННЯ ЗНАНЬ ТА УМІНЬ (НАЦІОНАЛЬНА ТА ECTS)

Таблиця 1 – Розподіл балів

	Поточний контроль			Семестровий контроль	Всього за семестр
	КР1	лб			
Підсумкові бали	80			20	100
Макс. проміжні бали	58	2			
Кільк. од. обліку у семестрі	1	11			
Макс. проміжних балів, всього	58	22			80
Коеф.. перерахунку	1				
Макс. кільк. підсумкових балів	58	22		20	100

Таблиця 2. Шкала оцінювання знань та умінь: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою
90 ... 100	A	відмінно
82 ... 89	B	добре
74 ... 81	C	
64 ... 73	D	задовільно
60 ... 63	E	
35 ... 59	FX	незадовільно з можливістю повторного складання
0 ... 34	F	незадовільно з обов'язковим повторним вивченням дисципліни

НАВЧАЛЬНО-МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Алгоритми і структури даних: практикум: навч. посіб./ Н.К. Стратієнко, М.Д. Годлевський, І.О. Бородіна.- Харків: НТУ"ХПИ", 2017. - 224 с.

2. Методичні вказівки до виконання лабораторних робіт з курсу "Алгоритми і структури даних" : для студ., які навч. за спец. 121 "Інженерія програмного забезпечення" [Електронний ресурс] / уклад. Н. К. Стратієнко, І. О. Бородіна ; Харківський політехнічний ін-т, нац. техн. ун-т. – Електрон. текстові дані. – Харків, 2017. – 36 с. – Режим доступу: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/26426>.

3. Методичні вказівки до виконання курсової роботи по курсу "Алгоритми та структури даних" : для студ., які навч. за напрямком 6.050103 "Програмна інженерія" спец. 05010301 "Програмне забезпечення систем" [Електронний ресурс] / уклад. Н. К. Стратієнко, О. В. Шматко, І. О. Бородіна ; Харківський політехнічний ін-т, нац. техн. ун-т. – Електрон. текстові дані. – Харків, 2016. – 28 с. – Режим доступу: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/24697>.

4. Guidance for course work on "Algorithms and Data Structures" : for students of direction 6.050103 "Software Engineering", specialty .05010302 "Software Engineering" = Методичні вказівки до виконання курсової роботи по курсу "Алгоритми та структури даних" : для студ., які навч. за напрямком 6.050103 "Програмна інженерія" спец. 05010302 "Інженерія програмного забезпечення" [Electronic resource] / comp. N. K. Stratiienko, O. V. Shmatko, I. O. Borodina ; National Technical University "Kharkiv Polytechnic Institute". – Electronic text data. – Kharkiv, 2016. – 28 p. – Access mode:

<http://repository.kpi.kharkov.ua/handle/KhPI-Press/24695>.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Базова

1. Алгоритми і структури даних: практикум: навч. посіб./ Н.К. Стратієнко, М.Д. Годлевський, І.О. Бородіна.- Харьков: НТУ"ХПИ", 2017. - 224с.
2. Н. Вирт. Алгоритмы и структуры данных. Новая версия для Оберона – М.: ДМК Пресс, 2010. –272 с.
3. А.В. Ахо, Д.Э.Хопкрофт, Д.Д.Ульман: Структуры данных и алгоритмы. — М. : Вильямс, 2003 – 384 с.
4. Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. Алгоритмы: построение и анализ.2-е изд./ Пер. с англ. под ред.А.Шеня. – М.: МЦНМО, 2005. – 1293 с.
5. Игошин В.И. Математическая логика и теория алгоритмов : учеб. пособие для студ. высш. учеб. заведений / В. И. Игошин. — 2-е изд., стер. — М. : Издательский центр «Академия», 2008. — 448 с.
6. Игошин В.И. Задачи и упражнения по математической логике и теории алгоритмов / В. И. Игошин. — 3-е изд., стер. — М. : Издательский центр «Академия», 2007. — 304 с.

Допоміжна література

7. Романовский И.В. Вычислительная математика и структура алгоритмов. — М.: МГУ. 2006. —112 с.
8. Поляков В.И., Скорубский В.И. Основы теории алгоритмов. – СПб: СПб НИУ ИТМО, 2012. – 51 с.
9. Рублев, В.С. Основы теории алгоритмов: учеб. пособие. / В.С. Рублев; Ярослав. гос. ун-т. – Ярославль: ЯрГУ, 2005. – 143 с.
10. Ульянов М.В. Ресурсно-эффективные компьютерные алгоритмы. Разработка и анализ: учебное пособие : М.: ФИЗМАТЛИТ, 2008. – 304 с.
11. Зайцева Е.В., Гурова Л.М. Математическая логика и теория алгоритмов: Учебное пособие. М.: МГГУ, 2006. – 255 с.

ІНФОРМАЦІЙНІ РЕСУРСИ В ІНТЕРНЕТІ

- 12 Архів комп'ютерної документації [Електронний ресурс]. – Режим доступу : infocity.kiev.ua/.