

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МИКОЛАЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ В. О. СУХОМЛИНСЬКОГО
Механіко-математичний факультет

Кафедра комп'ютерних наук та прикладної математики



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ

Ступінь бакалавра

Галузь знань 12 Інформаційні технології
спеціальність 122 «Комп'ютерні науки»
освітня програма Комп'ютерні науки

2019 – 2020 навчальний рік

Розробник: Ходякова Галина Вікторівна, доцент кафедри комп'ютерних наук та прикладної математики, кандидат педагогічних наук  (Ходякова Г.В.)

Робоча програма затверджена на засіданні кафедри комп'ютерних наук та прикладної математики

Протокол № 1 від «27» серпня 2019 р.

Завідувач кафедри  (Поздєєв В.О.)
«27» серпня 2019 р.

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, освітній ступінь	Характеристика навчальної дисципліни	
		<i>денна форма навчання</i>	
Кількість кредитів – 4	Галузь знань 12 Інформаційні технології	Нормативна	
	Спеціальність 122 Комп'ютерні науки		
Загальна кількість годин 120	Освітня програма: Комп'ютерні науки	<i>Рік підготовки:</i>	
		1-й	
		<i>Семестр</i>	
		1-й	
		<i>Лекції</i>	
Тижневих годин для денної форми навчання: аудиторних – 4 самостійної роботи студента – 6	Ступінь бакалавра	10 год.	
		<i>Практичні, семінарські</i>	
		<i>Лабораторні</i>	
		20 год.	
		<i>Самостійна робота</i>	
		90 год.	
		Вид контролю: іспит	

Мова навчання – українська.

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної та індивідуальної роботи становить: для денної форми навчання – 30 год. – аудиторні заняття, 90 год. – самостійна робота (30% ~ 70%).

2. Мета, завдання навчальної дисципліни та результати навчання

Мета викладання навчальної дисципліни «Алгоритмізація та програмування» є розкриття важливого значення алгоритмів як основ інформатики; ознайомлення з формальними методами, використовуваними для розробки та аналізу алгоритмів; демонстрація процесу розробки та аналізу алгоритмів; ознайомлення з набором фундаментальних концепцій побудови алгоритмів та структур даних; ознайомлення з класами алгоритмів, призначених для вирішення певного набору завдань; формування вміння оцінювати алгоритми, застосовувати різні алгоритми в залежності від контексту задачі; формування у студентів професійних знань, умінь та навичок, що дозволять їм оперативно обирати алгоритм, придатний для вирішення конкретного завдання або показувати, що такого алгоритму не існує; поліпшення професійної підготовки в області програмування і структур даних; ознайомлення студентів з передовим світовим досвідом у галузі програмування, алгоритмів вирішення задач, структур даних; надання студентам знань і досвіду використання алгоритмів і методів рішення, незалежно від мови програмування, парадигми програмування, апаратного забезпечення та інших аспектів реалізації алгоритмів і структур даних.

Завдання курсу «Алгоритмізація та програмування» є отримання студентом компетенцій для того, щоб приймати участь у проектуванні програм обробки інформації, створення складних програм різних типів з використанням складних структур даних і алгоритмів їх обробки.

- оволодіння основами алгоритмізації на рівні, достатньому для опрацювання математичних моделей, пов'язаних з подальшою практичною діяльністю фахівця;
- набуття умінь та навичок у сфері розробки програмного забезпечення;
- систематизація знань щодо використання алгоритмів;
- розмежування базових алгоритмічних структур та структур даних;
- вибір оптимальних алгоритмів розв'язку завдань;
- аналіз отриманих результатів та тестування власних алгоритмів;
- вивчення типових підходів до розробки і аналізу найбільш поширених алгоритмів розв'язку задач.

Студенти повинні знати :

- основні етапи процесу проектування програмного забезпечення;
- структури даних, що використовуються для подання типових інформаційних об'єктів;
- основні завдання аналізу алгоритмів;
- типові структури даних;
- складні структури даних;
- типові алгоритми обробки даних;
- основні методи розробки машинних алгоритмів і програм, структури даних, що використовуються для подання типових інформаційних об'єктів;

вміти:

- розробляти ефективні алгоритми на основі базових алгоритмічних структур;
- використовувати стандартні структури даних;
- створювати програмні реалізації типових алгоритмів;

- оцінювати ефективність алгоритмів обробки даних.
- реалізовувати алгоритми і використовувати структури даних засобами мов програмування високого рівня (на C++).
- розробляти алгоритми, використовуючи викладені в курсі загальні схеми, методи і прийоми побудови алгоритмів, вибираючи підходящі структури даних для представлення інформаційних об'єктів;
- доводити коректність складеного алгоритму і оцінювати основні характеристики його складності;
- реалізовувати алгоритми і використовувати структури даних засобами мов програмування високого рівня (на C++);
- експериментально (за допомогою комп'ютера) досліджувати ефективність алгоритму і програми.

Передумови для вивчення дисципліни: основи інформатики, дискретна математика, алгебра, математична логіка

Навчальна дисципліна складається з 4-х кредитів.

Очікувані результати навчання:

Студенти повинні застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.

Використовувати методологію системного аналізу об'єктів, процесів і систем для задач аналізу, прогнозування, управління та проектування процесів у прикладних задачах.

I. Загальнопредметні:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК17 Професійне володіння комп'ютером та інформаційними технологіями

II. Фахові:

КЗН-3 Знання фундаментальних розділів математики, в обсязі, необхідному для володіння математичним апаратом системних наук та кібернетики, здатність використовувати математичні методи в інформатиці.

КІ-6 Знання законів, методів та методик проведення наукових та прикладних досліджень.

КЗП-2 Знання математичних методів побудови та аналізу моделей природних, техногенних, економічних та соціальних об'єктів та процесів інформатизації, розробки математично обґрунтованих алгоритмів функціонування комп'ютеризованих систем (інформаційних систем, систем штучного інтелекту тощо).

КСП-4 Знання математичних методів розробки та дослідження алгоритмів розв'язування задач моделювання об'єктів і процесів інформатизації, алгоритмів функціонування інформаційних систем та методик оцінювання складових ефективності даних алгоритмів.

3. Програма навчальної дисципліни

Кредит 1. Універсальні структури даних

Тема 1. Функції користувача.

Структурний аналіз. Головна функція. Оголошення функції користувача. Формальні та фактичні параметри. Глобальні і локальні змінні. Опис функцій користувача. Виклик функції. Передача параметрів у функції. Основні поняття та термінологія, структур даних, вимоги до структур даних. Масиви і їх передача у функції.

Тема 2. Класифікація структур даних.

За складністю – прості та інтегровані. За способом представлення – фізичні та логічні. За наявністю зв'язків – зв'язні та незв'язні. За мінливістю – статичні, напівстатичні та динамічні. За впорядкуванням – лінійні і нелінійні. Структури даних для оперативної пам'яті та для зовнішньої пам'яті.

Кредит 2. Алгоритми пошуку і сортування

Тема 3. Алгоритми пошуку.

Алгоритм лінійного пошуку. Алгоритм двійкового пошуку. Алгоритми пошуку у текстах.

Тема 4. Алгоритми сортування.

Прямі методи сортування. Покращені алгоритми сортування. Удосконалені алгоритми сортування.

Кредит 3. Опрацювання текстових даних та файлів

Тема 5. Обробка текстових даних. Клас String

Обробка рядків: пошук, заміна, перестановка в рядках. Генерування тексту. Алгоритми пошуку в тексті. Специфіка комп'ютерної обробки літературної мови. Основи обробки текстів у прикладних областях: інформаційно-пошукові системи, електронні словники, переклад та ін.

Тема 6. Обробка файлів

Поняття файлу. Потоки даних. Виняткові ситуації, пов'язані з роботою з файлами. Основні алгоритми над файловими структурами. Текстові файли та їх обробка. Обробка файлів стандартизованих форматів.

Кредит 4. Стандартна бібліотека STL

Тема 7. Стандартна бібліотека шаблонів STL

Структура бібліотеки. Контейнери. Ітератори. Посилання. Типова організація класу контейнера. Розподільники, предикати і функтори.

Тема 8. Елементарні структури struct

Опис елементарних структур, операції із структурами. Програма-меню. Опрацювання структур.

4. Структура навчальної дисципліни

Назви кредитів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
		л	п	лаб	інд	с.р.
Кредит 1. Універсальні структури даних						
Тема 1. Функції користувача	14	2		2		10
Тема 2. Класифікація структур даних	16	2		2		12
Усього:	30	4		4		22
Кредит 2. Алгоритми пошуку і сортування						
Тема 3. Алгоритми пошуку	14	2		2		10
Тема 4. Алгоритми сортування	16			6		10
Усього:	30	2		8		20
Кредит 3. Опрацювання текстових даних та файлів						
Тема 5. Обробка текстових даних Клас String	14	2		2		10
Тема 6. Обробка файлів	16			2		14
Усього:	30	2		2		24
Кредит 4. Підпрограми.						
Тема 7. Стандартна бібліотека шаблонів STL	14	2		2		10
Тема 8. Елементарні структури struct	16			2		14
Усього:	30	2		4		24

Усього годин:	120	10		20		90
----------------------	------------	-----------	--	-----------	--	-----------

5. Теми лекційних занять

№ з/п	Назва теми	Кількість годин
	<i>Кредит 1. Універсальні структури даних</i>	
1	Тема 1. Функції користувача	2
2	Тема 2. Класифікація структур даних	2
	<i>Кредит 2. Алгоритми пошуку і сортування</i>	
3	Тема 3. Алгоритми пошуку. Алгоритми сортування	2
	<i>Кредит 3. Опрацювання текстових даних та файлів</i>	
4	Тема 4. Обробка текстових даних Клас String. Обробка файлів	2
	<i>Кредит 4. Підпрограми.</i>	
5	Тема 5. Стандартна бібліотека шаблонів STL	2
	Разом	10

6. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
	<i>Кредит 1. Універсальні структури даних</i>	
1	Тема 1. Передача параметрів у функції.	2
2	Тема 2. Одновимірні масиви у функціях.	2
3	Тема 3. Двовимірні динамічні масиви у функціях.	2
	<i>Кредит 2. Алгоритми пошуку і сортування</i>	
4	Тема 4. Опрацювання масивів у функціях	2
5	Тема 5. Пряме сортування масивів	2
6	Тема 6. Удосконалене сортування масивів	2

	<i>Кредит 3. Опрацювання текстових даних та файлів</i>	
7	Тема 7. Опрацювання текстових даних класом String	2
8	Тема 8. Підключення та робота з файлами	2
	<i>Кредит 4. Стандартна бібліотека STL</i>	
9	Тема 9. Застосування стандартних структур даних бібліотеки STL	2
10	Тема 10. Програмування структур та їх опрацювання	2
	Разом	20

7. Самостійна робота

№ з/п	Назва теми	Кількість годин
	<i>Кредит 1. Вступ до програмування. Базові поняття</i>	
1	Тема 1. Розв'язування задач на сайті e-olimp.	10
2	Тема 2. Оформлення звітів. Підготовка до захисту робіт	10
	<i>Кредит 2. Основні алгоритмічні структури</i>	
3	Тема 3. Розв'язування задач на сайті e-olimp.	10
4	Тема 4. Оформлення звітів. Підготовка до захисту робіт	18
	<i>Кредит 3. Стандартні складені структури даних.</i>	
5	Тема 5. Розв'язування задач на сайті e-olimp	10
6	Тема 6. Оформлення звітів. Підготовка до захисту робіт	12
	<i>Кредит 4. Підпрограми</i>	
7	Тема 7. Розв'язування задач на сайті e-olimp.	10
8	Тема 8. Оформлення звітів. Підготовка до захисту робіт	10
	Разом	90

8. Форми роботи та критерії оцінювання

Рейтинговий контроль знань студентів здійснюється за 100-бальною шкалою.
Шкала оцінювання: національна та ECTS

9. Засоби діагностики

Засобами діагностики та методами демонстрування результатів навчання є: завдання до лабораторних занять, завдання для самостійної та індивідуальної роботи, презентації результатів досліджень, тестові завдання, контрольні роботи.

10. Методи навчання

Курс складається з лекційних, лабораторних занять та самостійної роботи студентів, домашніх завдань і завершується підсумковим рейтинг-контролем і виконанням конкретних контрольних завдань по даній дисципліні.

Лекційні заняття призначені для теоретичного осмислення і узагальнення базових розділів курсу.

Лабораторні заняття є аудиторними, проводяться по наперед відомих темах у вигляді практичної роботи студентів за комп'ютерами. Лабораторне заняття – форма навчального заняття, спрямована на закріплення та вдосконалення студентом теоретичних знань, отриманих як на лекційних заняттях, так і в процесі самостійного вивчення матеріалу. Під час лабораторного заняття студенти під керівництвом викладача особисто набувають практичних навичок у роботі з обчислювальною технікою, оволодівають методикою створення програмних продуктів у програмному середовищі. У ході заняття студенти самостійно виконують передбачені завданням дії, заносючи результати в звіт. На це відводиться до 85 – 90% часу заняття. Викладач здійснює контроль за роботою і надає допомогу при виникненні труднощів, звертає увагу на складні, ключові моменти. Причому основна увага приділяється не вказівці на конкретну помилку, а методиці пошуку причин виникнення цих помилок. Складання звіту – це відповідальний етап лабораторного заняття. При його складанні студенти розвивають навички аналізу, узагальнення і творчого осмислення результатів роботи, а також навички розробки документації. Кожна лабораторна робота передбачає наявність індивідуального завдання і наступного захисту.

Самостійна робота є позааудиторною і призначена для самостійного ознайомлення студента з певними розділами курсу за рекомендованими педагогом матеріалами і підготовки до виконання індивідуальних завдань по курсу.

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.

Кінцевий контроль знань з дисципліни проводиться під час складання іспиту.

11. Методичне забезпечення

Бібліотека університету має достатню кількість підручників та посібників, лабораторних 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 лабораторних робіт та додаткові матеріали розміщуються на сервері кафедри та у навчально-методичному комплексі дисципліни.

12. Рекомендована література

Базова

1. Ашарина, И.В. Основы программирования на языках С и С++ / И.В. Ашарина. - М.: ГЛТ, 2012. - 208 с.
2. Вирт Н. Алгоритмы и структуры данных. - М.: Мир, 1989. - 360 с.
3. Страуступ, Б. Язык программирования С++. Специальное издание / Б. Страуступ. - М.: Бином, 2015. - 1136 с.

Допоміжна

1. Ахо А., Хопкрофт Д., Ульман Д. Структуры данных и алгоритмы. М.: Вильямс, 2000
2. М. Эллис, Б. Страуступ. Справочное руководство по языку С++ с комментариями: Пер. с англ. - Москва: Мир, 1992. 445с.
3. Стенли Б. Липпман. С++ для начинающих: Пер. с англ. 2тт. - Москва: Унитех; Рязань: Гэлион, 1992, 304-345сс.
4. Бруно Бабэ. Просто и ясно о Borland C++; Пер. с англ. - Москва: БИНОМ, 1994. 400с.
5. В. Липский Комбинаторика для программистов.-1988. 200 с.
6. С.Окулов Программирование в алгоритмах.- 2006. 220с
7. Серджвик Р. Фундаментальные алгоритмы на С.Часть 1-3 СПб: Диасофт, 2003.
8. Кениг, Э. Эффективное программирование на С++. Практическое программирование на примерах. Т. 2 / Э. Кениг, Б.Э. Му. - М.: Вильямс, 2016. - 368 с.
9. МакГрат, М. Программирование на С для начинающих / М. МакГрат. - М.: Эксмо, 2015. - 192 с.
10. Мартынов, Н.Н. Программирование для Windows на С / Н.Н. Мартынов. - М.: БИНОМ, 2013. - 528 с.
11. Перри, Г. Программирование на С для начинающих / Г. Перри, Д. Миллер. - М.: Эксмо, 2015. - 368 с.
12. 9. Фленов, М.Е. Программирование на С++ глазами хакера. / М.Е. Фленов. - СПб.: BHV, 2012. - 352 с.
13. Хенкеманс, Д. Программирование на С++ / Д. Хенкеманс, М. Ли. - СПб.: Символ-плюс, 2015. - 416 с.
14. Бочков С.О. Субботин Д.М. Язык программирования С для персонального компьютера. - М.: Радио и связь, 1990.
15. В.В. Подбельский. Язык С++: Учебное пособие. - Москва: Финансы и статистика, 1995. 560с.
16. Г. Шилдт. Самоучитель С++: Пер. с англ. - Санкт-Петербург: BHV-Санкт-Петербург, 1998. 620с.
17. У. Сэвитч. С++ в примерах: Пер. с англ. - Москва: ЭКОМ, 1997. 736с.
18. К. Джамса. Учимся программировать на языке С++: Пер. с англ. - Москва: Мир, 1997. 320с.
19. Х. Дейтел, П. Дейтел. Как программировать на С++: Пер. с англ. - Москва: ЗАО "Издательство БИНОМ", 1998. 1024с.
20. Керниган Б., Ритчи Д. Язык программирования С. - М.: Финансы и статистика, 1985.
21. Керниган Б., Ритчи Д., Фьюэр А. Язык программирования С. Задачи по языку С. - М.: Финансы и статистика, 1985.
22. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ, 2-е издание – М.: Вильямс, 2006.
23. Абельсон Х., Сассман Д. Структура и интерпретация компьютерных программ. М.: Добросвет, 2006
24. Уильям Топп, Уильям Форд. Структуры данных в С++: Пер. с англ. Топп, Уильям Форд. Структуры данных в С++: Пер. с англ. – М.: ЗАО «Издательство БИНОМ», 2000. – 816 с.

25. Караванова Т.П. Информатика. Методи побудови алгоритмів та їх аналіз. - К.:Генеза. 2007. 216 с.
26. Дорогов, В.Г. Основы программирования на языке С: Учебное пособие / В.Г. Дорогов, Е.Г. Дорогова; Под общ. ред. проф. Л.Г. Гагарина. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 224 с.
27. Хьюз, Камерон. Параллельное и распределенное программирование на С++ / Хьюз Камерон, Хьюз Трейси; пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 672 с.
28. Чан, Теренс. Системное программирование на С++ для UNIX. \ Чан Теренс.; пер. с англ. СПб: БХВ-Петербург, 1999 г.
29. Шоу, А. Логическое проектирование операционных систем. / А. Шоу; пер. с англ. – М.: Мир, 1981. – 360 с.
30. Эндрюс, Грегори Р. Основы многопоточного, параллельного и распределенного программирования \ Эндрюс, Грегори Р.; пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 512 с.

Інформаційні ресурси

1. <http://www.cplusplus.com/>
2. <http://www.e-olymp.com/>
3. <http://www.firststeps.ru/cbuilder/r.php?1>
4. <http://teacher.ucoz.net/index/lekci po s/0-4>
5. <http://object.newmail.ru/obj4.html>
6. <http://www.rsdn.ru/article/cpp/fastdelegate.xml>
7. <http://www.cyberguru.ru/programming/cpp/cpp-velvet-way-page76.html>

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МИКОЛАЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ В. О. СУХОМЛИНСЬКОГО**

Механіко-математичний факультет

Кафедра комп'ютерних наук та прикладної математики



**ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ**

Ступінь бакалавра

Галузь знань 12 Інформаційні технології

спеціальність 122 «Комп'ютерні науки»

освітня програма Комп'ютерні науки

Миколаїв – 2019

Програму розроблено та внесено: Миколаївський національний університет імені В. О. Сухомлинського

РОЗРОБНИК ПРОГРАМИ: Ходякова Галина Вікторівна, доцент кафедри комп'ютерних наук та прикладної математики, кандидат педагогічних наук.

Програму схвалено на засіданні кафедри комп'ютерних наук та прикладної математики

Протокол від «27» серпня 2019 року № 1

Завідувач кафедри комп'ютерних наук та прикладної математики (Поздєєв В.О.)



Програму погоджено навчально-методичною комісією механіко-математичного факультету

Протокол від «27» серпня 2019 року №

Голова навчально-методичної комісії



(Пархоменко О.Ю.)

Програму погоджено навчально-методичною комісією університету

Протокол від «27» серпня 2019 року № 14

Голова навчально-методичної комісії університету



(Кузнецова О.А.)

Програма вивчення нормативної дисципліни «Алгоритмізація та програмування» складена відповідно до освітнього ступеня бакалаврів спеціальності 122 Комп'ютерні науки.

Предметом вивчення алгоритми та структури даних, програмування на C++.

1. Мета та завдання навчальної дисципліни

Мета викладання навчальної дисципліни «Алгоритмізація та програмування» є розкриття важливого значення алгоритмів як основ інформатики; ознайомлення з формальними методами, використовуваними для розробки та аналізу алгоритмів; демонстрація процесу розробки та аналізу алгоритмів; ознайомлення з набором фундаментальних концепцій побудови алгоритмів та структур даних; ознайомлення з класами алгоритмів, призначених для вирішення певного набору завдань; формування вміння оцінювати алгоритми, застосовувати різні алгоритми в залежності від контексту задачі; формування у студентів професійних знань, умінь та навичок, що дозволять їм оперативно обирати алгоритм, придатний для вирішення конкретного завдання або показувати, що такого алгоритму не існує; поліпшення професійної підготовки в області програмування і структур даних; ознайомлення студентів з передовим світовим досвідом у галузі програмування, алгоритмів вирішення задач, структур даних; надання студентам знань і досвіду використання алгоритмів і методів рішення, незалежно від мови програмування, парадигми програмування, апаратного забезпечення та інших аспектів реалізації алгоритмів і структур даних.

Завдання курсу «Алгоритмізація та програмування» є отримання студентом компетенцій для того, щоб приймати участь у проектуванні програм обробки інформації, створення складних програм різних типів з використанням складних структур даних і алгоритмів їх обробки.

- оволодіння основами алгоритмізації на рівні, достатньому для опрацювання математичних моделей, пов'язаних з подальшою практичною діяльністю фахівця;
- набуття умінь та навичок у сфері розробки програмного забезпечення;
- систематизація знань щодо використання алгоритмів;
- розмежування базових алгоритмічних структур та структур даних;
- вибір оптимальних алгоритмів розв'язку завдань;
- аналіз отриманих результатів та тестування власних алгоритмів;
- вивчення типових підходів до розробки і аналізу найбільш поширених алгоритмів розв'язку задач.

Студенти повинні знати :

- основні етапи процесу проектування програмного забезпечення;
- структури даних, що використовуються для подання типових інформаційних об'єктів;
- основні завдання аналізу алгоритмів;
- типові структури даних;
- складні структури даних;
- типові алгоритми обробки даних;
- основні методи розробки машинних алгоритмів і програм, структури даних, що використовуються для подання типових інформаційних об'єктів;

вміти:

- розробляти ефективні алгоритми на основі базових алгоритмічних структур;
- використовувати стандартні структури даних;
- створювати програмні реалізації типових алгоритмів;
- оцінювати ефективність алгоритмів обробки даних.
- реалізовувати алгоритми і використовувати структури даних засобами мов програмування високого рівня (на C++).
- розробляти алгоритми, використовуючи викладені в курсі загальні схеми, методи і прийоми побудови алгоритмів, вибираючи підходящі структури даних для представлення інформаційних об'єктів;
- доводити коректність складеного алгоритму і оцінювати основні характеристики його складності;
- реалізовувати алгоритми і використовувати структури даних засобами мов програмування високого рівня (на C++);
- експериментально (за допомогою комп'ютера) досліджувати ефективність алгоритму і програми.

Передумови для вивчення дисципліни: основи інформатики, дискретна математика, алгебра, математична логіка

Очікувані результати навчання:

Студенти повинні застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.

Використовувати методологію системного аналізу об'єктів, процесів і систем для задач аналізу, прогнозування, управління та проектування процесів у прикладних задачах.

I. Загальнопредметні:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК17 Професійне володіння комп'ютером та інформаційними технологіями

II. Фахові:

КЗН-3 Знання фундаментальних розділів математики, в обсязі, необхідному для володіння математичним апаратом системних наук та кібернетики, здатність використовувати математичні методи в інформатиці.

КЗП-2 Знання математичних методів побудови та аналізу моделей природних, техногенних, економічних та соціальних об'єктів та процесів інформатизації, розробки математично обґрунтованих алгоритмів функціонування комп'ютеризованих систем (інформаційних систем, систем штучного інтелекту тощо).

КСП-4 Знання математичних методів розробки та дослідження алгоритмів розв'язування задач моделювання об'єктів і процесів інформатизації, алгоритмів функціонування інформаційних систем та методик оцінювання складових ефективності даних алгоритмів.

3. Інформаційний обсяг навчальної дисципліни

Кредит 1. Універсальні структури даних

Тема 1. Функції користувача.

Структурний аналіз. Головна функція. Оголошення функції користувача. Формальні та фактичні параметри. Глобальні і локальні змінні. Опис функцій користувача. Виклик функції. Передача параметрів у функції. Основні поняття та термінологія, структур даних, вимоги до структур даних. Масиви і їх передача у функції.

Тема 2. Класифікація структур даних.

За складністю – прості та інтегровані. За способом представлення – фізичні та логічні. За наявністю зв'язків – зв'язні та незв'язні. За мінливістю – статичні, напівстатичні та динамічні. За впорядкуванням – лінійні і нелінійні. Структури даних для оперативної пам'яті та для зовнішньої пам'яті.

Кредит 2. Алгоритми пошуку і сортування

Тема 3. Алгоритми пошуку.

Алгоритм лінійного пошуку. Алгоритм двійкового пошуку. Алгоритми пошуку у текстах.

Тема 4. Алгоритми сортування.

Прямі методи сортування. Покращені алгоритми сортування. Удосконалені алгоритми сортування.

Кредит 3. Опрацювання текстових даних та файлів

Тема 5. Обробка текстових даних. Клас String

Обробка рядків: пошук, заміна, перестановка в рядках. Генерування тексту. Алгоритми пошуку в тексті. Специфіка комп'ютерної обробки літературної мови. Основи обробки текстів у прикладних областях: інформаційно-пошукові системи, електронні словники, переклад та ін.

Тема 6. Обробка файлів

Поняття файлу. Потоки даних. Виняткові ситуації, пов'язані з роботою з файлами. Основні алгоритми над файловими структурами. Текстові файли та їх обробка. Обробка файлів стандартизованих форматів.

Кредит 4. Стандартна бібліотека STL

Тема 7. Стандартна бібліотека шаблонів STL

Структура бібліотеки. Контейнери. Ітератори. Посилання. Типова організація класу контейнера. Розподільники, предикати і функтори.

Тема 8. Елементарні структури struct

Опис елементарних структур, операції із структурами. Програма-меню. Опрацювання структур.

4. Рекомендована література

Базова:

1. Ашарина, И.В. Основы программирования на языках С и С++ / И.В. Ашарина. - М.: ГЛТ, 2012. - 208 с.
2. Вирт Н. Алгоритмы и структуры данных. - М.: Мир, 1989. - 360 с.
3. Страуступ, Б. Язык программирования С++. Специальное издание / Б. Страуступ. - М.: Бином, 2015. - 1136 с.

Допоміжна

1. Ахо А., Хопкрофт Д., Ульман Д. Структуры данных и алгоритмы. М.: Вильямс, 2000
2. М. Эллис, Б. Страуструп. Справочное руководство по языку С++ с комментариями: Пер. с англ. - Москва: Мир, 1992. 445с.
3. Стенли Б. Липпман. С++ для начинающих: Пер. с англ. 2тт. - Москва: Унитех; Рязань: Гэлион, 1992, 304-345сс.
4. Бруно Бабэ. Просто и ясно о Borland С++: Пер. с англ. - Москва: БИНОМ, 1994. 400с.
5. В. Липский Комбинаторика для программистов.-1988. 200 с.
6. С.Окулов Программирование в алгоритмах.- 2006. 220с
7. Серджвик Р. Фундаментальные алгоритмы на С.Часть 1-3 СПб: Диасофт, 2003.
8. Кениг, Э. Эффективное программирование на С++. Практическое программирование на примерах. Т. 2 / Э. Кениг, Б.Э. Му. - М.: Вильямс, 2016. - 368 с.
9. МакГрат, М. Программирование на С для начинающих / М. МакГрат. - М.: Эксмо, 2015. - 192 с.
10. Мартынов, Н.Н. Программирование для Windows на С / Н.Н. Мартынов. - М.: БИНОМ, 2013. - 528 с.
11. Перри, Г. Программирование на С для начинающих / Г. Перри, Д. Миллер. - М.: Эксмо, 2015. - 368 с.
12. 9. Фленов, М.Е. Программирование на С++ глазами хакера. / М.Е. Фленов. - СПб.: BHV, 2012. - 352 с.
13. Хенкеманс, Д. Программирование на С++ / Д. Хенкеманс, М. Ли. - СПб.: Символ-плюс, 2015. - 416 с.
14. Бочков С.О. Субботин Д.М. Язык программирования С для персонального компьютера. - М.: Радио и связь, 1990.
15. В.В. Подбельский. Язык С++: Учебное пособие. - Москва: Финансы и статистика, 1995. 560с.
16. Г. Шилдт. Самоучитель С++: Пер. с англ. - Санкт-Петербург: BHV-Санкт-Петербург, 1998. 620с.
17. У. Сэвитч. С++ в примерах: Пер. с англ. - Москва: ЭКОМ, 1997. 736с.
18. К. Джамса. Учимся программировать на языке С++: Пер. с англ. - Москва: Мир, 1997. 320с.
19. Х. Дейтел, П. Дейтел. Как программировать на С++: Пер. с англ. - Москва: ЗАО "Издательство БИНОМ", 1998. 1024с.
20. Керниган Б., Ритчи Д. Язык программирования С. - М.: Финансы и статистика, 1985.
21. Керниган Б., Ритчи Д., Фьюэр А. Язык программирования С. Задачи по языку С. - М.: Финансы и статистика, 1985.
22. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ, 2-е издание – М.: Вильямс, 2006.
23. Абельсон Х., Сассман Д. Структура и интерпретация компьютерных программ. М.: Добросвет, 2006

24. Уильям Топп, Уильям Форд. Структуры данных в С++: Пер. с англ. Уильям Топп, Уильям Форд. Структуры данных в С++: Пер. с англ. – М.: ЗАО «Издательство БИНОМ», 2000. – 816 с.
25. Караванова Т.П. Информатика. Методи побудови алгоритмів та їх аналіз. - К.:Генеза. 2007. 216 с.
26. Дорогов, В.Г. Основы программирования на языке С: Учебное пособие / В.Г. Дорогов, Е.Г. Дорогова; Под общ. ред. проф. Л.Г. Гагарина. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 224 с.
27. Хьюз, Камерон. Параллельное и распределенное программирование на С++ / Хьюз Камерон, Хьюз Трейси; пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 672 с.
28. Чан, Теренс. Системное программирование на С++ для UNIX. \ Чан Теренс.; пер. с англ. СПб: БХВ-Петербург, 1999 г.
29. Шоу, А. Логическое проектирование операционных систем. / А. Шоу; пер. с англ. – М.: Мир, 1981. – 360 с.
30. Эндрюс, Грегори Р. Основы многопоточного, параллельного и распределенного программирования \ Эндрюс, Грегори Р.; пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 512 с.

Інформаційні ресурси

1. <http://www.cplusplus.com/>
2. <http://www.e-olymp.com/>
3. <http://www.firststeps.ru/cbuilder/r.php?1>
4. <http://teacher.ucoz.net/index/lekci po s/0-4>
5. <http://object.newmail.ru/obj4.html>
6. <http://www.rsdn.ru/article/cpp/fastdelegate.xml>
7. <http://www.cyberguru.ru/programming/cpp/cpp-velvet-way-page76.html>

5. Форма підсумкового контролю успішності навчання: іспит

Засоби діагностики успішності навчання:

Лекційні та лабораторні заняття з використанням традиційних методів, інтерактивних методик та комп'ютерної техніки, індивідуальна робота, самостійна робота.

Поточний та підсумковий контроль здійснюється у вигляді комп'ютерних тестів на освітньому просторі університету. Для оцінювання використовується національна чотириохвальна шкала: відмінно, добре, задовільно, незадовільно; європейська шкала: А, В, С, D, E, FX, F.

(приклад для заліку) 100% балів студенти накопичують на заняттях та під час поточного і підсумкового контролю, що регламентується робочою програмою викладача. (приклад для іспиту) 60% балів студенти накопичують на заняттях та під час поточного контролю, що регламентується робочою програмою викладача, 40% балів студенти набирають на іспиті.