

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ
Н.Г. ЧЕРНЫШЕВСКОГО»
Факультет компьютерных наук и информационных технологий

УТВЕРЖДАЮ
Декал факультета КНиИТ
Миронов С.В.

"25" мая 2023 г.

**Рабочая программа дисциплины
Методы программирования**

Специальность
10.05.01 Компьютерная безопасность

Специализация
Математические методы защиты информации

Квалификация выпускника
Специалист по защите информации

Форма обучения
Очная

Саратов,
2023

Статус	ФИО	Подпись	Дата
Преподаватель-разработчик	Барулина М.А., Кондратова Ю.Н.	<i>Кондратова Ю.Н.</i>	25.05.23
Председатель НМК	Кондратова Ю.Н.	<i>Кондратова Ю.Н.</i>	25.05.23
Заведующий кафедрой	Кондратова Ю.Н.	<i>Кондратова Ю.Н.</i>	25.05.23
Специалист Учебного управления			

1. Цели освоения дисциплины

Целями освоения дисциплины Методы программирования являются знакомство с основными комбинаторными и теоретико-графовыми алгоритмами, а также способами их эффективной реализации и оценки сложности, знакомство с современными методами программирования.

2. Место дисциплины в структуре ООП

Данная учебная дисциплина относится к обязательной части Блока 1 «Дисциплины (Модули)» ООП и направлена на формирование у обучающихся общепрофессиональных компетенций.

Для изучения дисциплины необходимы компетенции, сформированные у обучающихся в результате освоения курсов «Информационные технологии и программирование», «Дискретная математика», «Математическая логика и теория алгоритмов», «Языки программирования».

Компетенции, сформированные у обучающихся в результате освоения данной дисциплины, необходимы для освоения дисциплин «Защита программ и данных», «Программно-аппаратные средства обеспечения информационной безопасности», «Защита информации от утечки по техническим каналам», при выполнении научно-исследовательской деятельности, при подготовке к процедуре защиты и защите выпускной квалификационной работы

3. Результаты обучения по дисциплине

Код и наименование компетенции	Код и наименование индикатора (индикаторов) достижения компетенции	Результаты обучения
ОПК-7. Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ.	ОПК-7.1.2 знает базовые структуры данных; основные алгоритмы сортировки и поиска данных, комбинаторные и теоретико-графовые алгоритмы; общие сведения о методах проектирования, документирования, разработки, тестирования и отладки программного обеспечения; ОПК-7.2.1 умеет работать с интегрированной средой разработки программного обеспечения; разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач; ОПК-7.2.2 умеет применять известные методы	Знать базовые структуры данных; основные алгоритмы сортировки и поиска данных, комбинаторные и теоретико-графовые алгоритмы; Уметь работать с интегрированной средой разработки программного обеспечения; разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач; применять известные методы программирования и возможности базового языка программирования для решения типовых профессиональных задач;

	<p>программирования и возможности базового языка программирования для решения типовых профессиональных задач;</p> <p>ОПК-7.3.1 владеет навыками разработки, документирования, тестирования и отладки программ;</p> <p>ОПК-7.3.2 владеет навыками разработки алгоритмов решения типовых профессиональных задач;</p> <p>ОПК-7.3.3 владеет навыками создания программ на языках высокого и низкого уровня, применения методов и инструментальных средств программирования для решения профессиональных задач, осуществления обоснованного выбора инструментария программирования и способов организации программ.</p>	<p>Владеть навыками разработки, документирования, тестирования и отладки программ; навыками разработки алгоритмов решения типовых профессиональных задач; навыками создания программ на языках высокого и низкого уровня, применения методов и инструментальных средств программирования для решения профессиональных задач, осуществления обоснованного выбора инструментария программирования и способов организации программ.</p>
<p>ОПК-13. Способен разрабатывать компоненты программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности.</p>	<p>ОПК-13.1.3 знает общие принципы построения и использования современных языков программирования высокого уровня; язык программирования высокого уровня (объектно-ориентированное программирование); язык ассемблера персонального компьютера;</p> <p>ОПК-13.1.4 знает современные технологии программирования; показатели качества программного обеспечения; базовые структуры данных; основные комбинаторные и</p>	<p>Знать язык программирования высокого уровня (объектно-ориентированное программирование); современные технологии программирования; показатели качества программного обеспечения; базовые структуры данных; основные комбинаторные и теоретико-графовые алгоритмы, а также способы их эффективной реализации и оценки вычислительной сложности;</p> <p>Уметь</p>

	<p>теоретико-графовые алгоритмы, а также способы их эффективной реализации и оценки вычислительной сложности;</p> <p>ОПК-13.2.3 умеет формализовать поставленную задачу; работать с интегрированными средами разработки программного обеспечения;</p> <p>ОПК-13.2.4 умеет формализовать поставленную задачу; разрабатывать эффективные алгоритмы и программы; проводить оценку вычислительной сложности алгоритма; планировать разработку сложного программного обеспечения;</p> <p>ОПК-13.3.3 владеет навыками разработки, отладки, документирования и тестирования программ; навыками использования инструментальных средств отладки и дизассемблирования программного кода;</p> <p>ОПК-13.3.4 владеет методами оценки качества готового программного обеспечения; навыками разработки алгоритмов для решения типовых профессиональных задач.</p>	<p>умеет формализовать поставленную задачу; работать с интегрированными средами разработки программного обеспечения; разрабатывать эффективные алгоритмы и программы; проводить оценку вычислительной сложности алгоритма; планировать разработку сложного программного обеспечения;</p> <p>Владеть навыками разработки, отладки, документирования и тестирования программ; навыками использования инструментальных средств отладки и дизассемблирования программного кода; методами оценки качества готового программного обеспечения; навыками разработки алгоритмов для решения типовых профессиональных задач.</p>
--	---	---

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 8 зачетных единиц, 324 часов.

№ п/п	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Икр	Ср	Формы текущего контроля успеваемости (по неделям семестра) Формы промежуточной аттестации (по семестрам)
				Лекции	Лабораторные занятия		Икр	Ср			
					Общая трудоемкость	Из них – практическая					
1	2	3	4	5	6	7	8		9		
5-ый семестр											
1	Введение	5	1	2	2	-					
2	Алгоритмы сортировки	5	2-5	8	8	-	0	10	Контроль выполнения лабораторных заданий		
3	Алгоритмы эффективного поиска	5	6-11	12	12	-	0	20	Контроль выполнения лабораторных заданий		
4	Алгоритмы вычислительной геометрии	5	12-13	4	4	-	0	10	Контроль выполнения лабораторных заданий Контрольная работа		
5	Хеш-функции, хеш-таблицы	5	14-15	4	4	-	0	15	Контроль выполнения лабораторных заданий		
6	Алгоритмы теории строк	5	16-18	6	6	-	2	15	Контроль выполнения лабораторных заданий		
	Промежуточная аттестация									Зачет	
	ИТОГО в 5-м семестре			36	36	-	2	70			
6-ый семестр											
1	Графы, графовые алгоритмы	6	1-7	14	16	-	2	20	Контроль выполнения лабораторных заданий		

№ п/п	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					ИКТ	СР	Формы текущего контроля успеваемости (по неделям семестра) Формы промежуточной аттестации (по семестрам)
				Лекции	Лабораторные занятия		ИКТ	СР			
					Общая трудоемкость	Из них – практическая					
1	2	3	4	5	6	7	8		9		
2	Алгоритмы на перестановках	6	8	2	4	-	-		12	Контроль выполнения лабораторных заданий	
3	Динамическое программирование	6	9- 11	6	6	-	-		10	Контроль выполнения лабораторных заданий Контрольная работа	
4	Параллельные алгоритмы	6	12- 13	4	2	-	-		14	Контроль выполнения лабораторных заданий	
5	Современные технологии программирования	6	14	2	2	-			10	Контроль выполнения лабораторных заданий	
6	Объектно- ориентированное программирование	6	15- 16	4	2	-	2		10	Контроль выполнения лабораторных заданий	
Промежуточная аттестация										Экзамен	
ИТОГО в 6-м семестре				32	32	-	4		76	36	
ВСЕГО				68	68	-	6		146	36	

Содержание дисциплины

5 семестр

Раздел «Введение».

Понятие «алгоритм», примеры. Временная сложность алгоритма. Значение тета-, омега-, о- нотаций (большие и малые). Свойства тета-, омега-, о- нотаций.

Раздел «Алгоритмы сортировки».

Задача о сортировке. Виды сортировок. Теорема о нижней оценке числа сравнений для сортировок. Сортировка слиянием. Структура данных «куча». Основные операции с кучей. Очереди с приоритетами. HeapSort. Быстрая сортировка. Общий вид быстрой сортировки. Варианты выбора опорного элемента. Операция partition. Анализ времени работы и используемой памяти. Сравнение с HeapSort. Алгоритм нахождения k-ой порядковой статистики. Линейные сортировки. CountSort. RadixSort. Примеры работы алгоритмов. Приложения сортировок: задачи на сортировки.

Раздел «Алгоритмы эффективного поиска».

Бинарный поиск. Назначения и идея алгоритма. Оценка количества действий. Функции стандартной библиотеки C++, основанные на бинарном поиске. Модификации: реализация `index_first_of`, `index_last_of` на основе стандартной библиотеки C++. Задача о `minimax` пути в графе. Реализации функций `binary_search`, `index_first_of`, `index_last_of` без использования функций стандартной библиотеки C++, основанных на бинарном поиске. Оптимизация бинарного поиска для лучшего использования кэша процессора. Дерево отрезков. Вещественнозначный бинарный поиск. Троичный (тернарный) поиск. Назначение и применимость. Реализация. Время работы. Приложения.

Раздел «Алгоритмы вычислительной геометрии».

Скалярное и векторное произведения векторов. Физический смысл и формулы вычисления. Примеры приложений: проверка взаимного расположения четырех точек, проверка многоугольника на выпуклость. Проверка пересечения двух отрезков. Алгоритмы нахождения выпуклой оболочки. Обзор методов. Просмотр Джавриса. Обход Грэхема. Задача о двух наиболее удаленных точках. Задача о нахождении пары ближайших точек.

Раздел «Хеш-функции, хеш-таблицы».

Хеш-функции. Виды хэш-функций (по области применения). Особенности каждого из видов. Хэш-таблицы. Основные операции с хэш-таблицами. Время работы хэш-таблиц. Методы разрешения коллизий. Алгоритм Луна.

Раздел «Алгоритмы теории строк».

Задача поиска образца в тексте. Два вида препроцессинга данных. Префикс-функция строки. Алгоритм нахождения префикс функции строки. Анализ, доказательство корректности. Алгоритм Кнута-Морриса-Пратта. Доказательство корректности и анализ. Суффиксные деревья. Простейшие алгоритмы построения суффиксных деревьев. Примеры.

6 семестр

Раздел «Графы, графовые алгоритмы».

Графы. Основные определения. Формулировка задачи SSSP. Дерево

кратчайших путей. Свойства. Алгоритм нахождения кратчайшего пути из s в t по дереву кратчайших путей из вершины s . Алгоритм BFS. Его анализ и свойства. Постановка задачи SSSP на взвешенных графах. Случаи возможных постановок задачи. Алгоритм Дейкстры. Инварианты выполнения алгоритма Дейкстры. Доказательство корректности. Анализ времени работы. Алгоритм Форда-Беллмана. Обоснование корректности, время работы. Алгоритм Флойда. Обоснование корректности, время работы. Условия применимости каждого из алгоритмов. Сравнение различных алгоритмов решения задачи SSSP. Краткое описание каждого из них, применимость, время работы, особенности. Примеры задач, сводящихся к SSSP. Алгоритм поиска в глубину. Лемма о скобочном вложении. Лемма о сером пути. Примеры работы алгоритма. Мосты и точки сочленения. Топологическая сортировка.

Раздел «Алгоритмы на перестановках».

Перестановки. Перестановки как преобразования. Произведение перестановок. Обратная перестановка. Группа перестановок. Алгоритмы перемножения и нахождения обратной перестановки. Граф перестановки. Порядок перестановки. Нахождение номера перестановки в лексикографическом упорядочении. Нахождение перестановки по ее номеру в лексикографическом упорядочении. Подсчет числа инволюций. Таблица инверсий перестановки. Алгоритм восстановления перестановки по таблице инверсий и наоборот.

Раздел «Динамическое программирование».

Динамическое программирование. Идеология. Оптимальность подзадач. Примеры.

Раздел «Параллельные алгоритмы».

Многопроцессорные системы. Средства описания параллельных алгоритмов; методы построения параллельных алгоритмов; параллельные алгоритмы сортировки данных; параллельные алгоритмы генерации псевдослучайных чисел.

Раздел «Современные технологии программирования» .

Тенденции в мире программирования. Парадигмы программирования. Концепции языков программирования. Разработка через тестирование. Switch-технология.

Раздел «Объектно-ориентированное программирование».

История. Основные понятия. Определение ООП и его основные концепции. Диаграммы UML. Особенности реализации. Критика ООП. Объектно-ориентированные языки.

На лабораторных занятиях студенты решают задачи, приведенные в Фонде оценочных средств (ФОС) дисциплины.

План лабораторных занятий

На лабораторных занятиях студенты должны решить следующие типовые задачи, приведенные в ФОС, по следующим темам

1. Алгоритмы сортировки

2. Алгоритмы эффективного поиска. Бинарный поиск
3. Алгоритмы вычислительной
4. геометрии
5. Хеш-функции, хеш-таблицы
6. Алгоритмы теории строк
7. Графы, графовые алгоритмы
8. Алгоритмы на перестановках
9. Динамическое программирование
10. Параллельные алгоритмы
11. Современные технологии программирования
12. Объектно-ориентированное программирование

5. Образовательные технологии, применяемые при освоении дисциплины

В рамках лекционных занятий предполагается разбор различных задач и ситуаций, обсуждение в малых группах, обсуждение с привлечением всей аудитории.

При проведении занятий по данному курсу используются следующие активные и интерактивные формы обучения: интерактивные средства визуализации учебного материала, мультимедийные технологии, дистанционная работа со студентами.

Примеры кейс- задания и проектные задачи приведены в фондах оценочных средства.

При обучении лиц с ограниченными возможностями здоровья и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуализации обучения, сопровождение тьюторами в образовательном пространстве. При этом основной формой организации учебного процесса является интегрированное обучение лиц с ограниченными возможностями здоровья, т.е. все студенты обучаются в смешенных группах, имеют возможность постоянно общаться со сверстниками, благодаря чему легче адаптируются в социуме.

6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

В рамках самостоятельной работы студенты осуществляют подготовку к лабораторным занятиям, контрольным работам, изучение материалов лекций, освоение дополнительного материала.

Фонд оценочных средств дисциплины включает в себя задания лабораторных занятий, примерные задания для контрольных работ, вопросы для проведения промежуточной аттестации (зачет и экзамен).

Фонд оценочных средств оформлен в качестве приложения к учебной рабочей программе дисциплины «Методы программирования».

7. Данные для учета успеваемости студентов в БАРС

Таблица 1.1 – Таблица максимальных баллов по видам учебной деятельности

1	2	3	4	5	6	7	8	9
Семестр	Лекции и	Лабораторные занятия	Практические занятия	Самостоятельная работа	Автоматизированное тестирование	Другие виды учебной деятельности	Промежуточная аттестация	Итого
5	10	30	0	10	0	10	40	100
6	10	30	0	10	0	10	40	100

Программа оценивания учебной деятельности студента

5 семестр

Лекции

Посещаемость, опрос, активность и др. за один семестр - от 0 до 10 баллов.

Лабораторные занятия

Контроль выполнения лабораторных заданий в течение одного семестра - от 0 до 30 баллов.

Практические занятия

Не предусмотрено.

Самостоятельная работа

Выполнение домашних работ в течении семестра - от 0 до 10 баллов.

Автоматизированное тестирование

Не предусмотрено.

Другие виды учебной деятельности

Выполнение контрольной работы - от 0 до 10 баллов.

Промежуточная аттестация

от 0 до 40 баллов.

Промежуточная аттестация представляет собой зачет.

При проведении промежуточной аттестации

ответ на «отлично» оценивается от 31 до 40 баллов;

ответ на «хорошо» оценивается от 21 до 30 баллов;

ответ на «удовлетворительно» оценивается от 11 до 20 баллов;

ответ на «неудовлетворительно» оценивается от 0 до 10 баллов.

Таким образом, максимально возможная сумма баллов за все виды учебной деятельности студента за 5 семестр по дисциплине «Методы программирования» составляет 100 баллов.

Таблица 2.1 - Таблица пересчета полученной студентом суммы баллов по дисциплине «Методы программирования» в оценку (зачет)

61 балл и более	«зачтено»
меньше 60 баллов	«не зачтено»

Программа оценивания учебной деятельности студента

6 семестр

Лекции

Посещаемость, опрос, активность и др. за один семестр - от 0 до 10 баллов.

Лабораторные занятия

Контроль выполнения лабораторных заданий в течение одного семестра - от 0 до 30 баллов.

Практические занятия

Не предусмотрено.

Самостоятельная работа

Выполнение домашних работ в течении семестра - от 0 до 10 баллов.

Автоматизированное тестирование

Не предусмотрено.

Другие виды учебной деятельности

Выполнение контрольной работы - от 0 до 10 баллов.

Промежуточная аттестация

от 0 до 40 баллов.

Промежуточная аттестация представляет собой экзамен.

При проведении промежуточной аттестации

ответ на «отлично» оценивается от 31 до 40 баллов;

ответ на «хорошо» оценивается от 21 до 30 баллов;

ответ на «удовлетворительно» оценивается от 11 до 20 баллов;

ответ на «неудовлетворительно» оценивается от 0 до 10 баллов.

Таким образом, максимально возможная сумма баллов за все виды учебной деятельности студента за 6 семестр по дисциплине «Методы программирования» составляет 100 баллов.

Таблица 2.2 - Таблица пересчета полученной студентом суммы баллов по дисциплине «Методы программирования» в оценку (экзамен)

90-100 баллов	«отлично»
76-89 баллов	«хорошо»
61-75 баллов	«удовлетворительно»
0-60 баллов	«неудовлетворительно»

8. Учебно-методическое и информационное обеспечение дисциплины.

а) литература:

1. Быкова, В. В. Комбинаторные алгоритмы. Множества, графы, коды : учебное пособие / Быкова В.В.. – Красноярск : Сибирский федеральный университет, 2015. – 152 с. – Текст: электронный. - URL: <https://www.iprbookshop.ru/84215.html>
2. Клейнберг, Ж. Алгоритмы : разработка и применение. Классика Computers Science / Ж. Клейнберг, Е. Тардос. - СПб. : Питер, 2016. - 800 с. — Текст : электронный – URL: <https://ibooks.ru/products/354377>
3. Колдаев, В. Д. Структуры и алгоритмы обработки данных : учебное пособие / В. Д. Колдаев. - Москва : РИОР : ИНФРА-М, 2020. - 296 с. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1054007>
4. Ковалевская, Е. В. Методы программирования: учебное пособие / Е. В. Ковалевская, Н. В. Комлева. — Москва : Евразийский открытый институт, 2011. — 320 с. — ISBN 978-5-374-00356-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/10784.html>
5. Объектно-ориентированное программирование на C++ : учебник / И. В. Баранова, С. Н. Баранов, И. В. Баженова [и др.]. — Красноярск : СФУ, 2019. — 288 с.— Текст: электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/157572>

б) программное обеспечение и Интернет-ресурсы:

- 1) Microsoft Office, Microsoft Windows
- 2) MS Visual Studio.



9. Материально-техническое обеспечение дисциплины

Для проведения лекционных занятий необходимы лекционная аудитория с возможностью демонстрации электронных презентаций при уровне освещения, достаточном для работы с конспектом.

Для проведения лабораторных занятий необходим компьютерный класс с необходимым программным обеспечением с доступом к Интернету.

Программа составлена в соответствии с требованиями ФГОС ВО по специальности 10.05.01 Компьютерная безопасность и специализации «Математические методы защиты информации» (квалификация «Специалист по защите информации»).

Авторы:

профессор кафедры математических основ информатики
и олимпиадного программирования, д. ф.-м.н.

М. А. Барулина

зав кафедрой математических
основ информатики и олимпиадного
программирования, к. ф.-м.н.

_____ Ю.Н.Кондратова

Программа одобрена на заседании кафедры МОИиОП от «25» мая 2023 года, протокол № 11.