

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования «Новосибирский национальный исследовательский
государственный университет» (Новосибирский государственный университет, НГУ)

Физический факультет



Согласовано, декан ФФ

Блинов В.Е.

2024 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование на языке С

направление подготовки: **03.04.01 Прикладные математика и физика**

Профиль: **Прикладные математика и физика. Информационные процессы и системы**

Форма обучения: очная

Семестр	Общий объем	Виды учебных занятий (в часах)				Промежуточная аттестация (в часах)				
		Контактная работа обучающихся с преподавателем			Самостоятельная работа, не включая период сессии	Самостоятельная подготовка к промежуточной аттестации	Контактная работа обучающихся с преподавателем			
		Лекции	Практические занятия	Лабораторные занятия			Консультации	Зачет	Дифференциро ванный зачет	Экзамен
1	2	3	4	5	6	7	8	9	10	11
2	108	16	48		36	6			2	
Всего 108 часов / 3 зачётные единицы, из них: - контактная работа 66 часов										
ПК-1										

Руководитель программы
д.ф.-м.н.

И. Б. Логашенко

Новосибирск, 2024

Содержание

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы	3
2. Место дисциплины в структуре образовательной программы	3
3. Трудоемкость дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося	3
4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий.....	4
5. Перечень учебной литературы	6
6. Перечень учебно-методических материалов по самостоятельной работе обучающихся..	6
7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	6
8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине	7
9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине	7
10. Оценочные средства для проведения текущего контроля и промежуточной аттестации по дисциплине.....	8
Приложение 1 Аннотация по дисциплине	
Приложение 2 Оценочные средства по дисциплине	

1. Перечень планируемых результатов обучения по дисциплине, соотнесённых с установленными в программе индикаторами достижения компетенций

Результаты освоения образовательной программы (компетенции)	Индикаторы	Результаты обучения по дисциплине
ПК-1 Способность осваивать и применять специализированные знания в области физико-математических и (или) естественных наук в своей профессиональной деятельности.	ПК 1.1 Применяет специализированные знания естественных и (или) физико-математических наук при решении поставленных задач в специализированной области своей профессиональной деятельности.	<p><u>Знать:</u> фундаментальные основы структурного подхода в программировании</p> <p><u>Уметь:</u> работать с динамическими структурами данных: массивами и списками</p> <p><u>Владеть:</u> принципами оценки временной сложности и потребления памяти по программному коду реализации алгоритма</p>

2. Место дисциплины в структуре образовательной программы

Дисциплина «Программирование на языке С» является одной из дисциплин по выбору по направлению подготовки **03.04.01 Прикладные математика и физика**. Курс «Программирование на языке С» развивает знания, умения и навыки, сформированные у обучающихся по результатам общей базовой подготовки в рамках программ бакалавриата. Дисциплина дает магистранту необходимые знания, навыки и предоставляет инструменты для выполнения проектов на стыке областей физики, математики и информационных технологий.

3. Трудоемкость дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося

Трудоемкость дисциплины – 3 з.е. (108 часов)

Форма промежуточной аттестации: 2 семестр – дифференцированный зачет

№	Вид деятельности	Семестр
		2
1	Лекции, час	16
2	Практические занятия, час	48
3	Лабораторные занятия, час	-
4	Занятия в контактной форме, час, из них	66

5	из них аудиторных занятий, час	64
6	в электронной форме, час	-
7	консультаций, час	-
8	промежуточная аттестация, час	2
9	Самостоятельная работа, час	42
10	Всего, час	108

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Лекции (16 часов)

Наименование темы и их содержание	Объем, час
Лекция 1. Основы структурного программирования. Подход к решению задач. Задачи и области применения программирования и языка С. Основные управляющие конструкции алгоритмических языков (оператор, последовательность, переменная, ветвление, цикл, вложенные циклы).	2
Лекция 2. Функции и массивы данных. Динамические массивы. Указатели Создание пользовательских функций. Локальные и глобальные переменные. Возвращение значений функциями. Передача параметров в функции.	2
Лекция 3. Динамические массивы. Указатели Выделение и освобождение памяти. Одномерные и многомерные массивы данных. Арифметика указателей. Особенности работы с массивами через арифметику указателей.	2
Лекция 4. Указатели и ссылки. Алгоритм сортировки пузырьком Ссылки. Передача параметров по ссылке. Константы. Алгоритмы сортировки. Алгоритм сортировки пузырьком и его оптимизации. Быстрые алгоритмы сортировки. Математическая сложность алгоритмов. Алгоритмы, работающие с предварительной информацией о массивах.	2
Лекция 5. Системы счисления данных. Позиционные и непозиционные системы счисления. Исторический экскурс. Двоичная и шестнадцатеричная системы как основа в информатике. Алгоритмы перевода между системами счисления.	2
Лекция 6. Пользовательские типы данных Символьный тип данных. Строки в языке Си как одномерный массив символов. Кодировки. Структурный тип данных. Объединение разнородных данных как основа структурного подхода в программировании.	2
Лекция 7. Списки структур данных Способы оперирования со структурными типами данных.	2

Односвязные и двусвязные списки данных. Кольцевой список и кольцевой список с барьером. Деревья. Бинарные деревья. Добавление и удаление элементов в бинарных деревьях. Поиск по бинарным деревьям.	
Лекция 8. <i>Работа с файлами. Контроль динамической памяти</i> Понятие файла и файловой системы. Текстовые, бинарные и исполняемые файлы. Файл как поток. Буферизация данных при записи в файл. MemoryLeaks как критичная ошибка для программ, предназначенных для научных экспериментов. Способы контроля утечек памяти.	2

Практические занятия (48 часов)

Содержание практического занятия	Объем, час
Занятие 1. Задачи и области применения программирования и языка С. Основы языка С. Структура программы. Типы данных. Ввод-вывод. Условный оператор IF. Цикл FOR. Форматирование кода.	3
Занятие 2. Системы счисления. Бинарные операции. Хранение данных в ПК.	3
Занятие 3. Точность и погрешность. Типы данных и их предельные значения (int, unsigned int, float, double).	3
Занятие 4. Форматированный вывод и работа с printf(). Округление и вывод заданного количества знаков.	3
Занятие 5. Структурный подход в программировании. Функции. Глобальные и локальные переменные. Передача параметров.	3
Занятие 6. Одномерные и многомерные массивы данных. Препроцессор и директива #define.	3
Занятие 7. Сортировка введенного динамического массива, реализация алгоритма сортировки пузырьком и его оптимизации.	3
Занятие 8. Реализация единой функции сортировки в соответствии с математической моделью, за счет выделения функций IfRepace() и Replace().	3
Занятие 9. Обсуждение особенностей выполнения заданий. Сравнение различных реализаций. Выявление достоинств и недостатков каждой реализации.	3
Занятие 10. Символьный тип данных и строки. Таблица ASCII.	3
Занятие 11. Случайные и псевдослучайные числа. Работа с генератором случайных чисел в С.	3
Занятие 12. Обработка текста. Алгоритмика применения правил русского языка на примере текста, введенного пользователем.	3
Занятие 13.	3

Вычисление определителя с матриц помощью рекурсивных функций.	
Занятие 14. Вычисление обратной матрицы, как транспонированной матрицы алгебраических дополнений.	3
Занятие 15. Методы контроля утечек памяти в программах.	3
Занятие 16. Способы оценки трудоемкости программ. Стандартные методы определения узких и сложных мест в программах. Профилировщики кода.	3

Проведение семинарских занятий осуществляется в форме практической подготовки, предусматривающей участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью в области информационных технологий, связанных с проведением научных и практических работ.

Самостоятельная работа студентов (42 часа)

Перечень занятий на СРС	Объем, час
Подготовка к практическим занятиям.	26
Изучение теоретического материала, не освещаемого на лекциях	10
Подготовка к дифференцированному зачету	6

5. Перечень учебной литературы

1. Двойнишников С.В., Лысаков К. Ф. Основы программирования (язык С): Учеб. пособие / Новосиб. гос. ун-т. Новосибирск: ИПЦ НГУ, 2018. – 158 с. (42 экз.)
2. Вирт, Никлаус Алгоритмы и структуры данных: с примерами на Паскале / Никлаус Вирт ; [пер. с англ. Д. Б. Подшивалова 2-е изд., испр.] Санкт-Петербург : Невский Диалект, 2007 350, [1] с. : ил. ; 20 см. (Библиотека программиста) ISBN 5-7940-0065-1 (28 экз.)

6. Перечень учебно-методических материалов по самостоятельной работе обучающихся

1. Д. Э. Кнут: Искусство Программирования. Том 1 Основные Алгоритмы, 1976
2. Д. Э. Кнут: Искусство Программирования. Том 2 Получисленные алгоритмы, 1976
3. Д. Э. Кнут: Искусство Программирования. Том 3 Сортировка и Поиск, 1976

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

7.1 Ресурсы сети Интернет

Для освоения дисциплины используются следующие ресурсы:

- электронная информационно-образовательная среда НГУ (ЭИОС);
- образовательные интернет-порталы;
- информационно-телекоммуникационная сеть Интернет;
- «Российская национальная платформа открытого образования» (<http://openedu.ru/>), edX (www.edx.org);
- Веб-страницы ведущих международных центров СИ.

Взаимодействие обучающегося с преподавателем (синхронное и (или) асинхронное) осуществляется через личный кабинет студента в ЭИОС, электронную почту.

7.2 Современные профессиональные базы данных:

- Реферативно-поисковая база данных Reaxys (Elsevier)
- Реферативно-библиографическая база данных Scopus (Elsevier)
- Реферативно-библиографическая база данных Scifinder (Chemical Abstracts Service)
- Библиометрическая база данных Web of Science Core Collection (Thomson Reuters Scientific LLC.)
- База данных полнотекстовых научных журналов JSTOR.
- Электронная библиотека диссертаций Российской государственной библиотеки (ЭБД РГБ)
- Электронные ресурсы российской научной библиотеки eLibrary.ru
- Электронные ресурсы издательства American Chemical Society (ACS)
- Электронные ресурсы издательства Annual Reviews
- Электронные ресурсы Freedom Collection издательства Elsevier
- Электронные ресурсы издательства The Royal Society of Chemistry (RSC)
- Электронные ресурсы издательства Wiley

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

8.1 Перечень программного обеспечения

Для обеспечения реализации дисциплины используется стандартный комплект программного обеспечения (ПО), включающий приложения для работы с документами и презентациями.

8.2 Информационные справочные системы

Не используются.

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Для реализации дисциплины используются специальные помещения:

- Учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля, промежуточной и итоговой аттестации;
- Помещения для самостоятельной работы обучающихся.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду НГУ.

Материально-техническое обеспечение образовательного процесса по дисциплине для обучающихся из числа лиц с ограниченными возможностями здоровья осуществляется согласно «Порядку организации и осуществления образовательной деятельности по образовательным программам для инвалидов и лиц с ограниченными возможностями здоровья в Новосибирском государственном университете».

Материально-техническое обеспечение образовательного процесса по дисциплине для обучающихся из числа лиц с ограниченными возможностями здоровья осуществляется согласно «Порядку организации и осуществления образовательной деятельности по образовательным программам для инвалидов и лиц с ограниченными возможностями здоровья в Новосибирском государственном университете».

10. Оценочные средства для проведения текущего контроля и промежуточной аттестации по дисциплине

Перечень результатов обучения по дисциплине и индикаторов их достижения представлен в разделе 1.

10.1 Порядок проведения текущего контроля и промежуточной аттестации по дисциплине

Текущий контроль успеваемости:

Текущий контроль успеваемости осуществляется контролем посещения занятий обучающимся и выполнения запланированных работ.

Промежуточная аттестация:

Для успешного прохождения курса обучающиеся должны продемонстрировать знания о принципах программирования и программному конструированию на языке программирования С.

Итоговая аттестация по дисциплине проводится в устной форме путем ответов на вопросы, освещаемые во время учебных занятий.

Окончательная оценка работы студента в течение семестра происходит на дифференцированном зачёте.

Вывод об уровне сформированности компетенций принимается преподавателем. Положительная оценка ставится, когда все компетенции освоены не ниже порогового уровня. Оценки «отлично», «хорошо», «удовлетворительно» означают успешное прохождение промежуточной аттестации.

Описание критериев и шкал оценивания индикаторов достижения результатов обучения по дисциплине «Программирование на языке С»

Таблица 10.1

Код компетенции	Индикатор	Результат обучения по дисциплине	Оценочное средство
ПК-1	ПК 1.1 Применяет специализированные знания естественных и (или) физико-математических наук	<u>Знать:</u> фундаментальные основы структурного подхода в программировании	Дифференцированный зачет.

	при решении поставленных задач в специализированной области своей профессиональной деятельности.	<u>Уметь:</u> работать с динамическими структурами данных: массивами и списками	Работа на семинарских занятиях при обсуждении типовых ошибок, затруднений, а также идей по решению задач рамках дисциплины.
		<u>Владеть:</u> принципами оценки временной сложности и потребления памяти по программному коду реализации алгоритма	Работа на семинарских занятиях при решении задач в рамках дисциплины.

Таблица 10.2

Критерии оценивания результатов обучения	Шкала оценивания
<p><u>Решение заданий:</u></p> <ul style="list-style-type: none"> – задание решено правильно, – работа оформлена аккуратно, четкие рисунки и чертежи, – осмысленность, логичность и аргументированность изложения материала, – точность и корректность применения терминов и понятий. <p>«Сдать задачу» означает объяснение хода её решения и, при необходимости, ответы на дополнительные вопросы преподавателя, имеющие принципиальное значение для данной дисциплины. Свободно и аргументированно отвечает на дополнительные вопросы. В ответах на вопросы преподавателя обучающийся мог допустить непринципиальные неточности.</p> <p><u>Дифференцированный зачет:</u></p> <ul style="list-style-type: none"> – самостоятельность, осмысленность, структурированность, логичность и аргументированность изложения материала, отсутствие затруднений в объяснении процессов и явлений, а также при формулировке собственных суждений, – точность и корректность применения терминов и понятий, – наличие исчерпывающих ответов на дополнительные вопросы. <p>При изложении ответа на вопрос(ы) преподавателя обучающийся мог допустить непринципиальные неточности.</p>	Отлично
<p><u>Решение заданий:</u></p> <ul style="list-style-type: none"> – задание решено правильно, – работа оформлена аккуратно, четкие рисунки и чертежи, – осмысленность, логичность и аргументированность изложения материала, наличие затруднений в формулировке собственных суждений, – точность и корректность применения терминов и понятий, при наличии незначительных ошибок. <p>«Сдать задачу» означает объяснение хода её решения и, при необходимости, ответы на дополнительные вопросы преподавателя, имеющие принципиальное значение для данной дисциплины. Отвечает на дополнительные вопросы. В ответах на вопросы преподавателя обучающийся мог допустить непринципиальные неточности.</p>	Хорошо

<p><u>Дифференцированный зачет:</u></p> <ul style="list-style-type: none"> – самостоятельность, осмысленность, структурированность, логичность и аргументированность изложения материала, наличие затруднений в объяснении отдельных процессов и явления, а также при формулировке собственных суждений, – точность и корректность применения терминов и понятий при наличии незначительных ошибок, – наличие полных ответов на дополнительные вопросы с возможным присутствием ошибок. 	
<p><u>Решение заданий:</u></p> <ul style="list-style-type: none"> – задание решено правильно, - работа оформлена неаккуратно – неосознанность и неосновательность выбранных методов анализа, – нет осмысленности в изложении материала, наличие ошибок в логике и аргументации, – корректность применения терминов и понятий, при наличии незначительных ошибок. <p>«Сдать задачу» означает объяснение хода её решения и, при необходимости, ответы на дополнительные вопросы преподавателя, имеющие принципиальное значение для данной дисциплины. При ответах на вопросы допускает ошибки</p> <p><u>Дифференцированный зачет:</u></p> <ul style="list-style-type: none"> – теоретический и фактический материал в слабой степени подкреплён ссылками на научную литературу и источники, – частичное понимание и неполное изложение причинно-следственных связей, – самостоятельность и осмысленность в изложении материала, наличие ошибок в логике и аргументации, в объяснении процессов и явлений, а также затруднений при формулировке собственных суждений, – корректность применения терминов и понятий, при наличии незначительных ошибок, – наличие неполных и/или содержащих существенные ошибки ответов на дополнительные вопросы. 	<p><i>Удовлетворительно</i></p>
<p><u>Решение заданий:</u></p> <ul style="list-style-type: none"> – задание решено неправильно, – компилятивное, неосмысленное, нелогичное и неаргументированное изложение материала, – грубые ошибки в применении терминов и понятий, <p>«Сдать задачу» означает объяснение хода её решения и, при необходимости, ответы на дополнительные вопросы преподавателя, имеющие принципиальное значение для данной дисциплины. На дополнительные вопросы не отвечает.</p> <p><u>Дифференцированный зачет:</u></p> <ul style="list-style-type: none"> – фрагментарное и недостаточное представление теоретического и фактического материала, не подкреплённое ссылками на научную литературу и источники, – непонимание причинно-следственных связей, – отсутствие осмысленности, структурированности, логичности и аргументированности в изложении материала, – грубые ошибки в применении терминов и понятий, – отсутствие ответов на дополнительные вопросы. 	<p><i>Неудовлетворительно</i></p>

10.3 Типовые контрольные задания и иные материалы, необходимые для оценки результатов обучения

Задачи для решения

1. **Программа решения квадратного уравнения (усвоение оператора ветвления и контроля типов).**
 - целочисленные коэффициенты задаются пользователем с клавиатуры
 - разветвление решения на случай линейного и квадратного уравнения
 - контроль корректности деления и извлечения квадратного корня
2. **Печать всех простых чисел, не превышающих N (усвоение вложенных операторов цикла)**
 - N задается пользователем с клавиатуры
 - точная реализация математического определения: «имеет 2 делителя»
 - собственная реализация проверки делимости через приведение типов
 - методы оптимизации вычисления: 0 делителей, ограничение цикла.
3. **Вычислить число π с заданной точностью (кол-во знаков после запятой), используя ряд Грегори**
 - вычисления суммы бесконечного ряда, не зная количества необходимых членов
 - собственная реализация правила округления
 - проверка критерия остановки счета
 - вывод количества потребовавшихся членов ряда
4. **Программа для решения уравнения вида $F(x) = 0$ методом Ньютона.**
 - Функция $F(x)$ задается преподавателем
 - Функция и ее производная задаются в тексте программы в виде отдельных функций
 $\text{float Func(float x)}$ и $\text{float dFunc(float x)}$
 - Начальное приближение и требуемая точность вычисления задаются с клавиатуры
 - Проверьте свой ответ, используя открытые ресурсы в Интернете
5. **Вычисление интеграла функции $F(x)$ методом трапеций**
 - Используется та же функция, что и задаче 4
 - Функция задается в тексте программы в виде отдельной функции $\text{float Func(float x)}$
 - шаг (количество разбиений) задается пользователем с консоли.
 - Оцените точность вычисления
 - Проверьте свой ответ, используя открытые ресурсы в Интернете
6. **Программа по вычислению максимума, минимума, среднего значения, среднеквадратичного отклонения во введенном статическом массиве**
 - максимальный размер массива задается `#define`, а размер – пользователем
 - хранение массива в виде глобальной переменной
 - целочисленные элементы массива задаются пользователем с клавиатуры
 - написание собственных функций, передача параметров, возвращение значений
 - используйте указатели и оператор `*` в функциях для расчёта минимума и максимума (Min, Max)
 - используйте массивы и оператор `[]` в функциях для расчёта среднего и среднекв. отклонения (Mean, RMS)
7. **Программа по вычислению максимума, минимума, среднего значения, среднеквадратичного отклонения во введенном динамическом массиве**
 - размер массива задается пользователем
 - целочисленные элементы массива задаются пользователем с клавиатуры

- выделение и освобождение памяти в функции main ()
- перепишите Min, Max из предыдущей задачи через [], а Mean, RMS – через *

8. *Сортировка введенного динамического массива*

- реализация алгоритма сортировки пузырьком и его оптимизации
- направление сортировки задает пользователь
- реализация единой функции сортировки в соответствии с математической моделью, за счет выделения функций Compare () и Replace ()
- реализуйте в программе несколько функций сравнения, для различных алгоритмов сортировки (по убыванию, по возрастанию, по абсолютному значению), и передайте указатель на соответствующую функцию сравнения в единую функцию сортировки

9. *Работа с матрицами 3x3*

- Структура Matrix_3x3 для хранения данных
- Данные хранятся в виде статического двумерного массива (float data [3][3])
- Функции вычисления суммы, произведения, определителя и обратной матрицы
- Исходная матрица заполняется целыми числами
- Добавьте возможность чтения матрицы из файла (название задает пользователь с экрана; можно использовать функцию из задачи 10)
- Результат операций выводится на экран и сохраняется в файл

10. *Работа с матрицами NxM*

- Структура Matrix_NxM для хранения данных
- Данные хранятся в виде динамического массива.
- Функция InitMatrix задает размер и выделяет память.
- Функция PrintMatrix выводит матрицу на экран в табличном виде
- Функции вычисления суммы, произведения, определителя и обратной матрицы
- Исходная матрица должна заполняться из файла или случайными числами
- Результат операций выводится на экран и сохраняется в файл
- Вычисление определителя с использованием рекурсивной функции (разложение по строке)

11. *Обработка текста, введенного пользователем*

- текст хранится в статическом массиве заданного размера (ограничение на размер вводимого текста)
- реализация функции коррекции текста:
 - 1) большая буква только в начале предложения или в начале текста,
 - 2) удаление нескольких пробелов, следующих подряд,
 - 3) удаление пробелов перед точкой и запятой,
 - 4) добавление пробела после знаков препинания.

12. *Реализация функции ввода текста произвольного размера: char* GetText(FILE* stream, char terminator)*

- Конец текста обозначается выбранным символом terminator (например, '#' или '\n')
- Возвращается строка в стиле C (нуль-терминированная), не включающая символа terminator

- Выделение точного количества необходимой памяти (собственная функция перераспределения памяти без использования `realloc()`)
- Использование функции коррективы из задачи 9
- Ввод текста из консоли или файла (задается с консоли)

13. *Записная книжка*

- Список из структур данных `SPeople`
- Поля данных: имя, фамилия, номер телефона, дата рождения
- Функции добавления элемента, удаления (по имени-фамилии), сортировки (по любому полю)
- Собственная реализация функции сравнения строк
- Реализация «пользовательского меню» (действие выбирает пользователь с консоли)
- Функция сохранения списка в файл и загрузки из файла (с добавлением либо заменой по выбору пользователя)

Вопросы к зачету

- Основы структурного программирования. Подход к решению задач
- Функции и массивы данных. Динамические массивы. Указатели
- Динамические массивы. Указатели
- Указатели и ссылки. Алгоритм сортировки пузырьком
- Системы счисления данных.
- Пользовательские типы данных
- Списки структур данных
- Работа с файлами. Контроль динамической памяти

Оценочные материалы по промежуточной аттестации, предназначенные для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС ВО, хранятся на кафедре-разработчике РПД в электронном виде.

Оценочные материалы по промежуточной аттестации (приложение 2), предназначенные для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС, хранятся на кафедре-разработчике РПД в печатном и электронном виде.

**Лист актуализации рабочей программы дисциплины
«Программирование на языке С»**

[illegible]

**Аннотация рабочей программы дисциплины
«Программирование на языке С»**

Программа дисциплины «Программирование на языке С» составлена в соответствии с ФГОС ВО по направлению подготовки **03.04.01 Прикладные математика и физика**, а также задачами, стоящими перед Новосибирским государственным университетом по реализации Программы развития НГУ. Дисциплина реализуется кафедрой автоматизации физико-технических исследований физического факультета Федерального государственного автономного образовательного учреждения высшего образования «Новосибирский национальный исследовательский государственный университет» (НГУ) для обучающихся магистратуры.

Цель дисциплины – получение базовых знаний о принципах программирования и обучение программному конструированию на языке программирования С.

Дисциплина нацелена на формирование у обучающихся универсальной компетенции:

Результаты освоения образовательной программы (компетенции)	Индикаторы	Результаты обучения по дисциплине
ПК-1 Способность осваивать и применять специализированные знания в области физико-математических и (или) естественных наук в своей профессиональной деятельности.	ПК 1.1 Применяет специализированные знания естественных и (или) физико-математических наук при решении поставленных задач в специализированной области своей профессиональной деятельности.	<p><u>Знать:</u> фундаментальные основы структурного подхода в программировании</p> <p><u>Уметь:</u> работать с динамическими структурами данных: массивами и списками</p> <p><u>Владеть:</u> принципами оценки временной сложности и потребления памяти по программному коду реализации алгоритма</p>

Курс рассчитан на один семестр. Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, практические занятия, самостоятельная работа студентов и дифференцированный зачет.

Программой дисциплины предусмотрены следующие виды контроля:

Текущий контроль успеваемости: работа на семинарских занятиях при решении задач в рамках дисциплины.

Промежуточная аттестация: дифференцированный зачёт.

Общая трудоемкость рабочей программы дисциплины составляет **108** академических часа / 3 зачетные единицы.