


МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

УТВЕРЖДАЮ

Руководитель ОПОП
Гензе Л.В.

" 31 " 08 
2021 г.

Рабочая программа дисциплины
Программирование

Закреплена за кафедрой	<i>Кафедра вычислительной математики и компьютерного моделирования</i>
Учебный план	<i>Математика 01.03.01, «Основы научно-исследовательской деятельности в области математики»</i>
Форма обучения	<i>очная</i>
Общая трудоёмкость	<i>13 з.е.</i>
Часов по учебному плану	<i>468 ч.</i>
в том числе:	
аудиторная контактная работа	<i>102 ч. лекции, 136 ч. практика, 12,4ч. групповые консультации</i>
самостоятельная работа	<i>163,6 ч.</i>
Вид(ы) контроля в семестрах	
<i>зачет</i>	<i>2,3 семестр</i>
<i>экзамен</i>	<i>1,4 семестр</i>

Томск-2021

Программу составили
профессор, доктор физико-математических наук

Старченко Александр Васильевич

Старший преподаватель наук
Гольдин Виктор Данилович

Рецензент
доцент, кандидат физико-математических наук

Берцун Владимир Николаевич

Рабочая программа дисциплины «Программирование» разработана в соответствии с СУОС
НИ ТГУ:

*Самостоятельно устанавливаемые образовательные стандарты НИ ТГУ по направлению
подготовки 01.03.01 – Математика (Утвержден Ученым советом НИ ТГУ, протокол от
27.03.2019 №03)*

Рабочая программа одобрена на заседании УМК ММФ

Протокол от 30 января 2020 № 1

1. Цель освоения дисциплины

Цель дисциплины: подготовка специалиста к эффективному использованию методов программирования в профессиональной деятельности при решении научно-практических задач.

Задачи:

- обучающиеся должны знать теоретические основы программирования, баз данных, обработки данных и знаний в сети интернет, синтаксиса языка программирования высокого уровня C++;
- обучающиеся должны уметь составлять и отлаживать программы на языке C++ в различных парадигмах программирования;
- обучающиеся должны владеть навыками объектно-ориентированного программирования, работы с базами данных, данными и знаниями в сети интернет, реализации базовых алгоритмов директивного и структурного программирования.

2. Место дисциплины в структуре ООП

Дисциплина относится к общепрофессиональному циклу, его вариативной части, модулю «Компьютерные науки».

Пререквизиты дисциплины – результаты ЕГЭ по информатике.

Постреквизиты дисциплины: дисциплины компьютерных наук, научно-исследовательская работа, выполнение и защита выпускной квалификационной работы.

3. Компетенции и результаты обучения, формируемые в результате освоения дисциплины

Таблица 1

Компетенция	Индикатор компетенции	Код и наименование результатов обучения (планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций)
ОПК-1 - Способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности.	ИОПК 1.1. Демонстрирует навыки работы с профессиональной литературой по основным естественнонаучным и математическим дисциплинам.	ОР-1.1 Может подбирать и обрабатывать профессиональную литературу по выбранной теме исследования.
	ИОПК 1.2 Демонстрирует навыки выполнения стандартных действий решения типовых задач с применением основных понятий и общих закономерностей, формулируемых в рамках базовых математических и естественнонаучных дисциплин.	ОР-1.2 Владеет фундаментальными знаниями в области программирования, такими как: - основные понятия и определения теоретических основ директивного, структурного и объектно-ориентированного программирования; основные алгоритмы поиска, сортировки. - базы данных, глобальная сеть, ее информационные ресурсы и структурирование.
	ИОПК 1.3 Владеет фундаментальными знаниями, по-	ОР-1.3 Умеет: правильно выбирать алгоритм для

	лученными в области математических и (или) естественных наук.	решения конкретной задачи, написать и отладить программу на языке C++ при решении практических задач; - осуществлять работу с базами данных, а также хорошо ориентироваться в глобальной сети, её информационных ресурсах и их структурировании; -осуществлять поиск необходимой литературы по программированию.
ОПК 2: - Способен разрабатывать, анализировать и внедрять новые математические модели в современных естествознании, технике, экономике и управлении	ИОПК 2.1: - Использует методы построения и анализа математических моделей в задачах естествознания, технике, экономике и управлении. ИОПК 2.2: - Демонстрирует умение применять на практике математические модели и компьютерные технологии для решения различных задач в области профессиональной деятельности. ИОПК 2.3:- Участвует в разработке математических моделей для решения задач естествознания, техники, экономики и управления под руководством более квалифицированного работника.	ОР-2.1 Обучающийся способен использовать методы построения и анализа математических моделей в задачах. ОР-2-2. Обучающийся умеет: - применять на практике математические модели и компьютерные технологии для решения конкретной задачи в области профессиональной деятельности ОР-2.3 Может участвовать в разработке математических моделей для решения задач исследовательского характера, выбирать и применять методы и современные компьютерные технологии под руководством более квалифицированного работника.

4. Структура и содержание дисциплины

4.1. Структура и трудоемкость видов учебной работы по дисциплине

Общая трудоемкость дисциплины составляет 13 зачетных единиц, 468 часов.

Таблица 2

Вид учебной работы	Трудоемкость в академических часах				
	1сем	2сем	3сем	4сем	всего
Общая трудоемкость					
Контактная работа:	77.9	50.65	56.95	69.5	255
Лекции (Л):	36	16	18	32	102
Практические занятия (П)	36	32	36	32	136
Групповые консультации	3.6	2.65	2.95	3.2	12.4
Промежуточная аттестация	18	0	0	18	36
Самостоятельная работа обучающегося:	14.4	57.35	51.05	40.8	163,6
Написание, компиляция и отладка программ на языке C++.	6.4	35.35	35.05	22.8	99.6
Изучение теоретических основ программирования	2	5	5	10	22
Подготовка отчетов по лабораторным работам.	6	17	11	10	44
Вид промежуточной аттестации	экз	зач	зач	экз	

4.2. Содержание и трудоемкость разделов дисциплины

Таблица 3

Код занятия	Наименование разделов и тем и их содержание	Вид учебной работы, занятий, контроля	Семестр	Часы в электронной форме	Всего (час.)	Контроль	Код (ы) результата(ов) обучения
	Тема 1. Теоретические основы программирования и алгоритмизации. Язык программирования С++.						
1.1	История развития вычислительной техники. Классы современных компьютеров. Архитектура фон Неймана. Принципиальное устройство компьютеров. Оперативная память. Процессор. Система элементарных команд процессора.	Л	1	0	4		ОР-1.1, ОР-1.2
1.2	Позиционные системы счисления. Алгоритмы перевода чисел из одной системы счисления в другую. Системы счисления с основанием 10, 2, 8, 16. Сложение и вычитание в произвольной системе счисления.	П	1	0	5	Контрольная работа	ОР-1.3
1.3	Понятие алгоритма, его свойства. Способы представления алгоритмов. Представление алгоритмов с помощью блок-схем.	Л	1	0	2		ОР-1.1, ОР-1.2
1.4	Блок схемы основных алгоритмов: последовательное выполнение, ветвление, циклические алгоритмы, работа с рекуррентными последовательностями	П	1	0	6	Контрольная работа	ОР-1.3, ОР-2.1, ОР-2.2
1.5	Структура программы на языке С (С++). Комментарии в программе, команды препроцессора. Понятие типа данных. Стандартные скалярные типы данных. Переменные в языке С (С++), их описание и использование; область действия переменных. Выражения. Оператор присваивания. Операторы ввода-вывода с использованием стандартных устройств в языках С, С++. Форматированный ввод-вывод в языке С. Пример простейшей программы	Л	1	0	5		ОР-1.2, ОР-1.3
1.6	Выполнение двух индивидуальных заданий по п.1.5	П	1	0	5	2 индивидуальных занятия	ОР-2.1, ОР-2.2,

							OP-2.3
1.7	Условный оператор. Оператор множественного выбора switch . Примеры программ, реализующих ветвящиеся алгоритмы	Л	1	0	2		OP-1.3
1.8	Операторы цикла с пред-, и постусловием. Примеры программ, реализующих циклические алгоритмы: алгоритмы суммирования, работа с рекуррентными последовательностями, оценка предельных значений	Л	1	0	2		OP 1.3
1.9	Выполнение семи индивидуальных заданий по п.1.7 и 1.8	П	1	0	10	7 индивидуальных заданий	OP-2.1, OP-2.2, OP-2.3
1.10	Тестирование и отладка программ. Приемы, часто применяемые при отладке. Основные принципы структурного программирования. Теорема Бомба-Джакопини. Метод проектирования программ «сверху вниз» - пошаговая детализация программ	Л	1	0	2		OP-1.1, OP-1.2
1.11	Логические утверждения как характеристика множества допустимых значений переменных. Элементы теории Т. Хоара : определение частичной правильности последовательности операторов, аксиома правильности для оператора присваивания, теорема о достаточном признаке частичной правильности цикла с предусловием. понятия инварианта цикла и ограничивающей функции	Л	1	0	2		OP-1.1, OP-1.2
1.12	Примеры проектирования программ с одновременным доказательством их правильности: быстрое возведение в степень, нахождение наибольшего общего делителя двух натуральных чисел (алгоритм Евклида).	Л	1	0	2		OP-1.1, OP-1.2
1.13	Массивы как тип данных. Одномерные, двумерные массивы. Оператор цикла for .	Л	1	0	2		OP-1.1, OP-1.2
1.14	Указатели и их использование. Динамические переменные. Эквивалентность массивов и указателей. Динамические массивы.	Л	1	0	2		OP-1.1, OP-1.2
1.15	Массивы как тип данных. Одномерные, двумерные массивы. Оператор цикла for . Динамические массивы	П	1	0	10	13 индивидуальных заданий	OP-2.1, OP-2.2, OP-2.3
1.16	Строки как тип данных и операции с ними (язык С). Ввод-вывод строковых переменных. Структуры как тип данных	Л	1	0	2		OP-1.1, OP-1.2

1.17	Понятие подпрограммы. Описание, вызов подпрограмм. Виды формальных параметров: входные, результаты, модифицируемые. Описание функции в языке С (С++). Формальные параметры функций. Вызов функции в языке С (С++). Действия компьютера при вызове функции. Соответствие формальных и фактических параметров при вызове функций. Глобальные параметры.	Л	1	0	2		ОР-1.1, ОР-1.2
1.18	Использование указателей и ссылок в качестве параметров функций. Использование массивов в качестве параметров функций. Указатели на функции.	Л	1	0	2		ОР-1.1, ОР-1.2
1.19	Прототипы функций. Заголовочные файлы и их подключение к проекту. Перегрузка функций. Формальные параметры функций по умолчанию	Л	1	0	2		ОР-1.2
1.20	Работа с файлами в языке С. Стандартные функции fscanf , fprintf . Работа с файлами в языке С++.	Л	1	0	3		ОР-1.2
1.21	Написание и отладка программ по индивидуальным заданиям. Подготовка отчетов.	Ср	1	0	14,4		ОР-2.1, ОР-2.2, ОР-2.3
	Тема 2. Структуры данных, строки, функции, файлы на языке С++. Алгоритмы поиска, сортировки, хеширования, оценка их эффективности и сложности						
2.1	Массивы как тип данных. Работа с двумерными массивами	П	2	0	6	8 индивидуальных заданий	ОР-1.2, ОР-2.1, ОР-2.2, ОР-2.3
2.2	Строки как тип данных и операции с ними.	П	2	0	4	Индивидуальное задание	ОР-1.2, ОР-2.1, ОР-2.2, ОР-2.3
2.3	Структуры как тип данных	П	2	0	2	Индивидуальное задание	ОР-1.2, ОР-2.1, ОР-2.2, ОР-2.3
2.4	Работа с файлами	П	2	0	4	2 индивидуальных задания	ОР-1.2, ОР-2.1, ОР-2.2, ОР-2.3
2.5	Использование функций. Использование указателей и ссылок в качестве параметров функций.	П	2	0	8	Индивидуальное задание	ОР-1.2, ОР-2.1, ОР-2.2, ОР-2.3
2.6	Использование массивов в качестве параметров функций	П	2	0	4	Индивидуальное задание	ОР-1.2, ОР-2.1, ОР-2.2, ОР-2.3
2.7	Рекурсивные функции. Примеры использования.. Правила, которые нужно соблюдать при написании рекурсивных функций. Реализация рекурсивных функций в компьютере	Л	2	0	2		ОР-1.1, ОР-1.2,

2.8	Программирование рекурсивных функций на C++	П	2	0	4	Индивидуальное задание	OP-1.2, OP-2.1, OP-2.2
2.9	Алгоритм поиска с возвратом на примере поиска пути в дорожной сети. Эффективность и сложность алгоритмов. Классификация алгоритмов по их сложности.	Л	2	0	3		OP-1.2
2.10	Работа с информацией, расположенной в таблице (одномерном массиве). Постановка задач поиска, вставки и удаления элементов таблицы. Алгоритмы поиска, вставки и удаления в информации в неупорядоченном массиве. Оценка их эффективности. Алгоритм бинарного поиска в упорядоченном массиве; оценка его эффективности. Алгоритмы вставки и удаления информации в упорядоченном массиве, оценка их эффективности	Л	2	0	4		OP-1.2
2.11	Постановка задачи сортировки одномерного массива. Алгоритмы сортировки: метод «пузырька», метод простого включения, метод прямого выбора, метод быстрой сортировки. Оценка их эффективности. Повышение эффективности стандартных методов сортировки	Л	2	0	5		OP-1.2
2.12	Хэширование как метод размещения данных в массиве. Коллизии и способы их разрешения. Алгоритмы поиска и вставки информации в массив при использовании хэширования. Преимущества и недостатки технологии хэширования	Л	2	0	2		OP-1.2
2.13	Написание и отладка программы, подготовка отчета по индивидуальному заданию. Изучение теоретического материала	Ср	2	0	57,35		OP-1.1, OP-1.2, OP-2.1, OP-2.2
Тема 3. Динамические структуры данных: стек, очередь, список, деревья. Объектно-ориентированное программирование							
3.1	Алгоритм поиска с возвратом	П	3	0	4	Индивидуальное задание	OP-2.1, OP-2.2, OP-2.3
3.2	Стек как структура данных. Логическое описание. Варианты физической реализации стека: на базе массива и с помощью динамических переменных. Функции, реализующие операции со стеком Очередь как структура данных. Логическое опи-	Л	3	0	2		OP-1.2

	вание. Варианты реализации очереди с помощью массива и динамических переменных. Функции, реализующие операции с очередью						
3.3	Программная реализация стека и очереди и функций для работы с ними	П	3	0	8	2 индивидуальных задания	OP-2.1, OP-2.2, OP-2.3
3.4	Линейный однонаправленный список (ЛОС), логическое описание. Варианты реализации списка с помощью массива и динамических переменных. Функции, реализующие операции с ЛОС	Л	3	0	1		OP-1.2
3.5	Программная реализация ЛОС и функций для работы с ним	П	3	0	2	Индивидуальное задание	OP-2.1, OP-2.2, OP-2.3
3.6	Деревья как структуры данных. Основные определения и способы реализации. Обходы деревьев. Деревья бинарного поиска. Алгоритмы поиска, вставки и удаления элементов в бинарном дереве поиска; оценка их эффективности	Л	3	0	3		OP-1.2
3.7	Программная реализация бинарных деревьев и функций для работы с ними	П	3	0	2	Индивидуальное задание	OP-2.1, OP-2.2, OP-2.3
3.8	Начала объектно-ориентированного программирования (ООП). Классы как обобщение понятия структуры. Инкапсуляция – соединение переменных и методов работы с ними в одной структуре. Описание класса в языке C++. Конструкторы классов; конструкторы по умолчанию, конструкторы копирования	Л	3	0	2		OP-1.2
3.9	Пример создания класса fraction, реализующего работу с рациональными числами. Определение новых операций для работы с объектами классов. Пример описания арифметических операций для рациональных чисел	Л	3	0	2		OP-1.2
3.10	Программная реализация класса fraction и операций для работы с объектами класса	П	3	0	4		OP-2.1, OP-2.2, OP-2.3
3.11	Шаблоны классов. Примеры создания классов для реализации стека, очереди и ЛОС.	Л	3	0	2		OP-1.2
3.12	Шаблоны классов. Примеры создания классов для реализации стека, очереди и ЛОС.	П	3	0	6	3 индивидуальных задания	OP-2.1, OP-2.2, OP-2.3
3.13	Использование принципа наследования при создании иерархии классов.	Л	3	0	1		OP-1.2
3.14	Стандартные классы из библиотеки C++: строки,	Л	3	0	2		OP-1.2

	потоки, контейнеры						
3.15	Решения индивидуальных задач с использованием собственных классов	П	3	0	8	4 индивидуальных задания	OP-2.1, OP-2.2, OP-2.3
3.16	Понятие операционной системы, основные элементы операционной системы, принципы взаимодействия прикладных программ с операционными системами. Исключения как классы, обеспечивающие взаимодействие программы с операционной системой	Л	3	0	3		OP-1.2
3.17	Исключения как классы, обеспечивающие взаимодействие программы с операционной системой	П	3	0	2	Индивидуальное задание	OP-2.1, OP-2.2, OP-2.3
3.18	Написание и отладка программы, подготовка отчета по индивидуальному заданию. Изучение теоретического материала	Ср	3	0	51,05		OP-1.1, OP-1.2, OP-2.1, OP-2.2, OP-2.3
	Тема 4 Базы данных.						
4.1	Базы данных : необходимость в особой технологии, основные определения. Архитектура баз данных. Этапы проектирования базы данных. Инфологическая модель данных: понятия сущности, атрибутов, связи, ключа, E/R –диаграммы Даталогические модели данных. Типы баз данных: иерархическая, реляционная, сетевая. Варианты организации хранения базы данных во внешней памяти и связанные с ними алгоритмы поиска и обновления данных: хэширование, индексированные файлы, Б – деревья.	Л	4	0	5		OP-1.2
4.2	Реляционная модель данных: основные определения. Реляционная алгебра. Нормализация.	Л	4	0	2		OP-1.2
4.3	Знакомство с базами данных на компьютере	П	4	0	2		OP-1.1, OP-1.2, OP-2.1, OP-2.2, OP-2.3
4.4	Язык запросов SQL. Типы данных в SQL. Операторы CREATE, INSERT, DELETE, UPDATE, SELECT.	Л	4	0	3		OP-1.2
4.5	Арифметические и логические операции, используемые в операторах SQL, агрегатные функции. Неопределенные значения данных и трехзначная логика. Реализация операций реляционной алгебры	П	4	0	4		OP-1.1, OP-1.2, OP-2.1, OP-2.2, OP-2.3

	ры в языке SQL.					
4.6	Реализация ограничений целостности в SQL. Триггеры и хранимые процедуры Встроенный SQL, взаимодействие SQL-операторов с программами, написанными на алгоритмическом языке высокого уровня. Динамические SQL-запросы Проблемы безопасности БД. Основные методы защиты данных. Понятие транзакции	Л	4	0	4	OP-1.2
4.7	Пример простой базы данных, создание инфологической и реляционной модели данных, нормализация. СУБД MS Access. Работа с базами данных в MS Access.	Л	4	0	2	OP-1.2
4.8	Проектирование конкретной базы данных по индивидуальному заданию: создание инфологической модели, нормализация. Реализация разработанной базы данных в MS Access	П	4	0	8	OP-1.1, OP-1.2, OP-2.1, OP-2.2, OP-2.3
4.9	Защита разработанного проекта	П	4	0	2	OP-1.1, OP-1.2, OP-2.1, OP-2.2, OP-2.3
	Тема 5. Глобальная сеть, её информационные ресурсы и их структурирование.					
5.1	Язык гипертекстовой разметки HTML	Л	4	0	4	OP-1.2
5.2	Введение в расширенный язык разметки XML	Л	4	0	4	OP-1.2
5.3	XML-схема	Л	4	0	4	OP-1.2
5.4	Создание XML-схем	Л	4	0	4	OP-1.2
5.5	Редактор для построения XML-схем. Построение простых схем и документов. Пример: схема для описания паспорта.	П	4	0	4	OP-1.1, OP-1.2, OP-2.2
5.6	Построение схемы для описания родственных отношений.	П	4	0	8	OP-1.1, OP-1.2, OP-2.2

5.7	XML- схема предметной области	П	4	0	4		OP-1.1, OP-1.2, OP-2.2
	Раздел 6. Консультация						
6.1	Консультация	КРто	1,2,3,4	0	12,4		OP-1.1, OP-1.2, OP-1.3, OP-2.1, OP-2.2, OP-2.3
	Раздел 7. Промежуточная аттестация						
7.1	Приём экзаменов и зачётов по дисциплине	КРто	1,2,3,4	0	4,6		OP-1.1, OP-1.2, OP-1.3, OP-2.1, OP-2.2, OP-2.3

5. Образовательные технологии, учебно-методическое и информационное обеспечение для освоения дисциплины

В ходе реализации дисциплины используются классические образовательные технологии – лекции, практические занятия; самостоятельное изучение рекомендованной литературы и постепенное выполнение индивидуального задания; промежуточная аттестация в приема зачетов (2, 3 семестры) и экзаменов (1,4 семестры).

Самостоятельная работа включает в себя: теоретическое освоение лекционного курса, практическое выполнение контрольных заданий и индивидуальных заданий, подготовку к зачету и экзамену. Для выполнения самостоятельной работы обеспечивается доступ к информационным ресурсам курса:

- материалы лекций;
- список вопросов для самостоятельной проверки знаний и подготовки к экзамену.
- список литературы, включающий учебники и книги по изучаемым в курсе вопросам.

Все контрольные работы и индивидуальные задания подобраны так, чтобы максимально стимулировать психологическую установку студентов-математиков на формирование связи между математической теорией и ее практическим применением. Отчет по каждой лабораторной работе включает теоретическую часть, выполненное практическое задание и анализ полученных результатов.

5.1. Литература и учебно-методическое обеспечение

а) Перечень основной учебной литературы.

1. Керниган Б., Ритчи Д. Язык программирования Си. СПб.: «Невский диалект», 2001.
2. Т. А. Павловская. С, С++. Программирование на языке высокого уровня. СПб.: ПИТЕР, 2002.
3. В. В. Подбельский Язык С++. М.: Финансы и статистика, 1999.
4. Т. А. Павловская, Ю. А. Щупак. С, С++. Структурное программирование: Практикум. СПб.:ПИТЕР, 2002.
5. Васильев А.Н. Программирование на С++ в примерах и задачах.
6. Конова Е., Поллак Г. Алгоритмы и программы. С++. СПб.: Издательство «Лань», 2017.384 с.
7. Ульман Дж. Д., Уидом Дж. Введение с системы баз данных. - М.: Издательство «Лори», 2000.
8. Ю.Б. Бакаревич, Н.В.Пушкина. Самоучитель MS Office 2016. – СПб., БХВ-Петербург, 2017, 418 с

б) Перечень дополнительной учебной литературы.

1. Вопросы и ответы по С и С++. Москва: «МикроАрт», 1997.
2. Брайан Оверленд. С++ без страха. М.; изд-во «Триумф», 2005.
3. Йодан Э. Структурное программирование и конструирование программ. М.:Мир, 1979.
4. Дейкстра. Дисциплина программирования. М.: Мир, 1978.
5. Вирт Н. Алгоритмы и структуры данных. М: Мир, 1985.
6. Мейер В., Бодуэн К. Методы программирования. М.:Мир,1982, Т.1,2.

7. Могилев А.В., Пак Н.И., Хеннер Е.К. Информатика. М.- Академия, 1999.
8. Дейт К. Дж. Введение в системы баз данных. – К.; М.; СПб.: Издательский дом «Вильямс», 1999
9. Задачник по программированию для математиков. Томск: Изд-во Том. ун-та, 2001.
10. Страуструп Б. Программирование. Принципы и практика использования С++. Москва: Вильямс, 2011 – 1238 стр.
11. Эдди С.Э. XML. Справочник, СПб., Питер, 1999, 480с.

5.2. Базы данных и информационно-справочные системы, в том числе зарубежные

1. ЭБС Лань <http://e.lanbook.com/books>
2. Открытый университет Интуит.ру <http://intuit.ru>;
3. <http://www.intuit.ru/studies/courses/2260/156/info>
4. <http://www.intuit.ru/studies/courses/66/66/info>
5. Богословский Н.Н. Программирование на С++ Электронный ресурс: учебно-методический комплекс. Томск : [ИДО ТГУ] , 2015 URL: <http://moodle.tsu.ru/course/view.php?id=431>
6. Расширяемый язык разметки (XML) 1.0 (вторая редакция). <http://citforum.ru/internet/xmlspec/index.shtml>
7. Пространства имен в XML. <http://citforum.ru/internet/xnamsps/index.shtml>
8. Язык XML - практическое введение, Александр Печерский <http://citforum.ru/internet/xml/index.shtml>

5.3. Перечень лицензионного и программного обеспечения

- 1) операционная система Windows 7 или Windows 10 <https://www.microsoft.com/ru-ru/software-download/windows10>
- 2) интегрированная среда для работы на С/С++.

5.4. Оборудование и технические средства обучения

Для проведения лабораторных работ и самостоятельной работы используются аудитории учебно-вычислительной лаборатории ММФ. При выполнении индивидуальных заданий, самостоятельных и лабораторных работ используется свободное и лицензионное программное обеспечение:

- офисный пакет Microsoft Office 2010 (составление отчетов).

6. Методические указания обучающимся по освоению дисциплины

Для успешного освоения материала студентам необходимо посещать занятия, а во время самостоятельной работы пользоваться основной и дополнительной литературой, базами данных и информационно-справочными системами, которые представлены в списке литературы. Самостоятельная работа студентов состоит в повторении материала с практических занятий и самостоятельного изучения дополнительных вопросов, более глубокого анализа темы с помощью литературы.

7. Преподавательский состав, реализующий дисциплину

Старченко Александр Васильевич, профессор, доктор физико-математических наук.
Гольдин Виктор Данилович, старший преподаватель
Зюзьков Валентин Михайлович, доцент

Лаева Валентина Ивановна, старший преподаватель
Данилкин Евгений Александрович, доцент
Шельмина Елена Александровна, доцент
Фазлиев Александр Зарипович, доцент
Михайлов Михаил Дмитриевич, старший преподаватель

8. Язык преподавания

Русский

Лист актуализации

Рабочей программы дисциплины «Программирование»

Программа Основы научно-исследовательской деятельности в области математики

Направление 01.03.01 Математика

Раздел (подраздел), в ко- торый вносятся	Основания для из- менений	Краткая характери- стика вносимых из- менений	Дата и номер про- токола заседания учебно-