Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования «Северо-Осетинский государственный университет

имени Коста Левановича Хетагурова»

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«**Основы и методология программирования**»

Направление подготовки:

**01.03.02 Прикладная математика и информатика**

Профиль: **«Программирование, анализ данных**

**и математическое моделирование»**

Квалификация (степень) выпускника – бакалавр

Форма обучения – очная

Год начала подготовки 2024

Владикавказ

2024

Рабочая программа

*обсуждена и утверждена* на заседании кафедры прикладной математики и информатики (протокол № 6 от 27.02.2024 г.);

*одобрена* советом факультета математики и компьютерных наук (протокол № 6 от 01.03.2024 г.);

*утверждена* в составе Основной профессиональной образовательной программы понаправлению подготовки 01.03.02 «Прикладная математика и информатика», профиль: «Программирование, анализ данных и математическое моделирование», год начала подготовки 2024 (решение ученого совета от 28.03.2024, протокол № 8).

Составители: доцент кафедры прикладной математики и информатики, к.ф.-м.н. Гутнова А.К., старший преподаватель кафедры прикладной математики и информатики Макаренко М.Д.

**1. Структура и общая трудоемкость дисциплины**

Общая трудоемкость дисциплины составляет 11 з.е. (396 ч.).

|  |  |
| --- | --- |
|  | Очная форма обучения |
| Курс | 1/2 |
| Семестр | 1/2/3/4 |
| Лекции | - |
| Практические занятия | - |
| Лабораторные занятия | 68/68/34/34 |
| Консультации |  |
| Итого аудиторных занятий | 68/68/34/34 |
| Самостоятельная работа | 22/22/74/74 |
| Курсовая работа | - |
| Зачет | - |
| Экзамен | - |
| Общее количество часов | 396 |

**2. Цели изучения дисциплины**

Целью освоения дисциплины «Основы и методология программирования» является получение фундаментальных знаний в области алгоритмизации, включающих: структурирование простых и сложных данных с использованием статических и динамических структур; методы доступа к данным; основные типы алгоритмов; способы проведение оценки сложности алгоритмов; методы создания эффективных алгоритмов; знакомство с классическими алгоритмами сортировки и поиска.

В задачи курса входит изучение студентами методов программирования для овладения знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования.

**3. Место дисциплины в структуре ОПОП:**

Б1.O.16. Блок 1. Дисциплины (модули). Обязательная часть.

Приступая к изучению дисциплины, студент должен иметь представление о понимании проблематики, целей и задач программирования, о современных технологиях программирования (структурное, модульное программирование).

**4. Требования к результатам освоения дисциплины**

В результате освоения дисциплины обучающийся должен обладать следующими компетенциями (результатами освоения образовательной программы):

Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1);

Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде (УК-3);

Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач (ОПК-2);

Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения (ОПК-5).

Взаимосвязь планируемых результатов обучения по дисциплине с формируемыми компетенциями ОПОП:

|  |  |  |  |
| --- | --- | --- | --- |
| **Код и формулировка компетенции** | **Планируемые результаты обучения, соответствующие формируемым компетенциям ОПОП** | | |
| **Знать:** | **Уметь** | **Владеть:** |
| УК-1  Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач | Знает базовые принципы по защите информации на рабочем месте и при входе в локальные и глобальные сети | Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности | Владеет навыками составления обзоров, рефератов и библиографии по тематике исследований |
| УК-3  Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде | Знает базовые принципы по защите информации на рабочем месте и при входе в локальные и глобальные сети | Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности | Владеет навыками составления обзоров, рефератов и библиографии по тематике исследований |
| ОПК-2  Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач | Знает базовые принципы по защите информации на рабочем месте и при входе в локальные и глобальные сети | Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности | Владеет навыками составления обзоров, рефератов и библиографии по тематике исследований |
| ОПК-5  Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения | Знает базовые принципы по защите информации на рабочем месте и при входе в локальные и глобальные сети | Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности | Владеет навыками составления обзоров, рефератов и библиографии по тематике исследований |

**5. Содержание и учебно-методическая карта дисциплины**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **№ темы** | **Наименование тем (вопросов), изучаемых по данной дисциплине** | **Занятия** | | **Самостоятельная работа студентов** | | **Формы контроля** | **Литература** |
| **л** | **лаб.** | **Содержание** | **Часы** |
| **1 семестр** | | | | | | | |
|  | Операторы ввода/вывода. Консольный ввод и вывод строк и чисел. |  | 2 |  |  | Проверка выполненных заданий | [1-10] |
|  | Основные базовые и структурированные типы данных, их характеристика. Условные переходы. |  | 4 | Составной оператор. Вложенные условные операторы. | 4 | Проверка выполненных заданий | [1-10] |
|  | Циклические переходы. |  | 12 | Циклы с предусловием While. | 4 | Проверка выполненных заданий | [1-10] |
|  | Основные понятия программирования. Алгоритм. Основные свойства алгоритма. Алгоритмизация задач. Типовые структуры алгоритма. Языки программирования. Основные элементы языка. Основы С# (идентификаторы, переменные, выражения, операции, операторы, атрибуты, спецификаторы). Типы данных (значимые, ссылочные, преобразование, упаковка / распаковка). |  | 4 |  |  | Проверка выполненных заданий | [1-10] |
|  | Стандартные и пользовательские типы данных (символьные, логические, целые, вещественные, структуры, перечисления, массивы, строки, классы, свойства, методы). |  | 8 | Побитовые операции | 4 | Проверка выполненных заданий | [1-10] |
|  | Функции. Основы решения алгоритмической задачи. Делимость, НОД, НОК. Рекурсия. |  | 4 | Локальные функции | 4 | Проверка выполненных заданий | [1-10] |
|  | Массивы одномерные. Типовые алгоритмы обработки массивов. Массивы как объекты. |  | 6 | Стандартные алгоритмы сортировки одномерных массивов | 6 | Проверка выполненных заданий | [1-10] |
|  | Строковый тип данных. |  | 12 |  |  | Проверка выполненных заданий | [1-10] |
|  | Двумерные массивы. Ступенчатые (“зубчатые”) массивы. Типовые  алгоритмы обработки двумерных массивов. |  | 16 |  |  | Проверка выполненных заданий | [1-10] |
|  | **Итого** | - | 68 |  | 22 |  |  |
| **2 семестр** | | | | | | | |
|  | Организация ввода-вывода данных. Работа с файлами. |  | 12 | Работа с  файлами | 4 | Проверка выполненных заданий | [1-10] |
|  | Визуализация. |  | 20 | Визуализация | 10 | Проверка выполненных заданий | [1-10] |
|  | Жизненный цикл программного обеспечения. Типы приложений. |  | 4 |  |  | Проверка выполненных заданий | [1-10] |
|  | Объектно-ориентированная парадигма: понятия объекта, класса объектов; основные понятия объектно-ориентированного программировании (инкапсуляция, наследование и полиморфизм); классы и объекты; интерфейсы и реализация. Делегаты и события. |  | 14 | История развития ООП. Классы объектов. Компоненты и их свойства. | 2 | Проверка выполненных заданий | [1-10] |
|  | Структуры. Стандартные списки, очереди. |  | 10 | Реализация динамических структур | 6 | Проверка выполненных заданий | [1-10] |
|  | Стеки, коллекция Stack. |  | 8 |  |  | Проверка выполненных заданий | [1-10] |
|  | **Итого** | - | 68 |  | 22 |  |  |
| **3 семестр** | | | | | | | |
|  | Технология создания программного кода. Технологии коллективной разработки программного обеспечения. Технологические средства разработки программного обеспечения. Методы отладки и тестирования программ. Категории программных ошибок. |  | 2 | Методы отладки и тестирования программ. Категории программных ошибок. | 10 | Проверка выполненных заданий | [1-10] |
|  | Основы анализа эффективности алгоритмов. Асимптотические обозначения и как их применяют для анализа эффективности алгоритмов. |  | 2 | Составление эффективных алгоритмов | 20 | Проверка выполненных заданий | [1-10] |
|  | Системный стек. Анализ сложности рекурсивных алгоритмов. |  | 4 |  |  | Проверка выполненных заданий | [1-10] |
|  | Сортировка. Основные задачи сортировки. Устойчивая и неустойчивая сортировки. |  | 4 | Отработка задач на сортировку данных | 20 | Проверка выполненных заданий | [1-10] |
|  | Алгоритм сортировки слиянием. |  | 4 |  |  | Проверка выполненных заданий | [1-10] |
|  | Алгоритма пузырьковой сортировки и ее  временная сложность. Разные алгоритмы в C#. |  | 8 |  |  | Проверка выполненных заданий | [1-10] |
|  | Алгоритм бинарного поиска. Метод быстрой сортировки. «Жадные алгоритмы»  Алгоритм «Поиск в глубину» и «Поиска в  ширину» |  | 10 | Жадный алгоритм раскраски графа. Сравнение эффективности работы жадных алгоритмов и рекурсивных. | 24 | Проверка выполненных заданий | [1-10] |
|  | **Итого** | - | 34 |  | 74 |  |  |
| **4 семестр** | | | | | | | |
|  | Дополнительные возможности ООП в C#. Определение операторов. Перегрузка операций преобразования типов. Индексаторы. |  | 2 | Переменные-ссылки и возвращение ссылки. Методы расширения. Частичные классы и методы. Анонимные типы. Кортежи. Records. | 8 | Проверка выполненных заданий | [1-10] |
|  | Pattern matching. Паттерн типов и свойств, паттерны кортежей, позиционный паттерн, реляционный и логический паттерны, паттерны списков |  | 6 |  |  | Проверка выполненных заданий | [1-10] |
|  | Многопоточность. Введение в многопоточность. Класс Thread. Создание потоков. Делегат ThreadStart. Потоки с параметрами и ParameterizedThreadStart. Синхронизация потоков. Мониторы. Класс AutoResetEvent. Мьютексы. Семафоры |  | 4 | Параллельное программирование и библиотека TPL. Задачи и класс Task. Работа с классом Task. Задачи продолжения. | 12 | Проверка выполненных заданий | [1-10] |
|  | Основы LINQ.  Проекция данных. Фильтрация коллекции. Сортировка. Объединение, пересечение и разность коллекций. Агрегатные операции. Получение части коллекции. Группировка. Соединение коллекций. Проверка наличия и получение элементов. Отложенное и немедленное выполнение LINQ. |  | 4 | Делегаты в запросах LINQ | 18 | Проверка выполненных заданий | [1-10] |
|  | Сборка мусора, управление памятью и указатели. Сборщик мусора в C#. Финализируемые объекты. Метод Dispose. Конструкция using. Указатели. Указатели на структуры, члены классов и массивы. |  | 4 |  |  |  | [1-10] |
|  | Работа с JSON.  Сериализация в JSON. JsonSerializer |  | 8 | Настройка сериализации с помощью атрибутов | 16 |  | [1-10] |
|  | Работа с XML в C#. XML-Документы. Работа с XML с помощью классов System.Xml. Изменение XML-документа. XPath. Linq to Xml. Создание Xml-документа. Выборка элементов в LINQ to XML. Изменение XML-документа в LINQ to XML. |  | 6 | Сериализация в XML. XmlSerializer | 20 |  | [1-10] |
|  | **Итого** | - | 34 |  | 74 |  |  |

**6. Образовательные технологии**

Согласно учебному плану при преподавании дисциплины используются традиционные образовательные технологии: лабораторные занятия и самостоятельная работа студентов. Также при проведении занятий и самостоятельной работе студентов могут быть использованы:

**–** *интерактивные технологии* («мозговой штурм», дебаты, презентационный метод, работа в парах, работа в группах, деловая игра);

**–** *технологии контекстного обучения* **–** система дидактических форм, методов и средств, направленная на моделирование содержания будущей профессиональной деятельности специалиста (анализ конкретных ситуаций, методы работы с информационными базами данных, деловая игра и др.);

– *технологии электронного обучения* (реализуется при помощи электронной образовательной среды СОГУ при использовании ресурсов ЭБС, при проведении автоматизированного тестирования и т. д.).

Все виды учебной работы могут проводиться дистанционно на основании локальных нормативных актов СОГУ.

В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по индивидуальной траектории в рамках индивидуального рабочего плана, изучение данной дисциплины может осуществляться через индивидуальные консультации преподавателя очно, в часы консультаций, по электронной почте или с использованием ЭИОС СОГУ.

**7. Учебно-методическое обеспечение самостоятельной работы**

Самостоятельная работа проводится с целью:

– систематизации и закрепления полученных теоретических знаний и практических умений обучающихся студентов;

– углубления и расширения теоретических знаний;

– формирования умений использовать нормативную, правовую, справочную документацию и специальную литературу;

– формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;

– развития исследовательских навыков и умений.

По дисциплине предусмотрены следующие виды самостоятельной работы студентов:

– самостоятельное повторение и изучение теоретического материала;

– выполнение домашних заданий;

– подготовка доклада/конспекта по теме, вынесенной на самостоятельное изучение;

– подготовка к выполнению лабораторных работ;

– подготовка к промежуточной аттестации (зачету).

Содержание, трудоемкость и формы контроля внеаудиторной самостоятельной работы содержатся в разделе 5.

**8. Оценочные средства для текущего контроля успеваемости,**

**рубежной и промежуточной аттестации по итогам освоения дисциплины**

Основными *формами текущего контроля* по дисциплине являются:

– проект по теме, вынесенной на самостоятельное изучение;

– проверка домашних заданий, лабораторных работ.

Форма *рубежного контроля:* контрольная работа.

Формы *промежуточной аттестации:* зачет.

**8.1. Формы контроля и критерии оценивания**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Этап** | **Форма**  **контроля** | **Критерии оценивания (процент от максимального кол-ва баллов)** | | | |
| **86-100 %** | **71–85%** | **50–70%** | **Менее 50%** |
| **отлично / зачет** | **хорошо / зачет** | **удовлетворительно / зачет** | **неудовлетворительно / незачет** |
| *1. Текущий контроль (max 20 баллов за один модуль)* | | | | | |
|  |  | 15–17 баллов | 12–14 баллов | 8–11 баллов | 0–7 баллов |
|  | Текущая работа в течение модуля  (мах 17б.) | Студент активно работает на занятиях, превосходно выполняет все задания преподавателя. | Студент активно работает на занятиях, хорошо выполняет задания преподавателя. | Студент недостаточно активно работает на занятиях, удовлетворительно выполняет задания преподавателя. | Студент недостаточно активно работает на занятиях, неудовлетворительно выполняет задания преподавателя. |
|  |  | 3 балла | 2 балла | 1 балл | 0 баллов |
|  | Реферат /  доклад / конспект  (мах 3б.) | Тема полностью раскрыта. Превосходное владение материалом. Высокий уровень самостоятельности, логичности, аргументированности. Превосходный стиль изложения. | Тема в основном раскрыта. Хорошее владение материалом. Средний уровень самостоятельности, логичности, аргументированности. Хороший стиль изложения. | Тема частично раскрыта. Удовлетворительное владение материалом. Низкий уровень самостоятельности, логичности, аргументированности. Удовлетворительный стиль изложения. | Тема не раскрыта. Неудовлетворительное владение материалом. Недостаточный уровень самостоятельности, логичности, аргументированности. Неудовлетворительный стиль изложения. |
| *2. Рубежный контроль (15 б. за один модуль)* | | | | | |
|  | тест /  контрольная работа | Количество баллов за выполнение каждого задания указываются в тесте / контрольной работе. | | | |
| *3. Промежуточная аттестация по дисциплине (max число баллов – в соответствии с действующим локальным нормативным актом)* | | | | | |
|  |  | Критерии оценивания (процент от максимального кол-ва баллов) | | | |
|  |  | 86–100 % | 71–85 % | 50–70 % | 0–49 % |
|  | Зачет /  Экзамен | Дан полный, развернутый ответ на поставленный вопрос. Ответ формулируется в терминах науки, изложен литературным языком, логичен, доказателен, демонстрирует авторскую позицию студента. | Дан полный ответ на поставленный вопрос, показано умение выделить существенные и несущественные признаки, причинно-следственные связи. Но допущены незначительные ошибки, исправленные студентом с помощью «наводящих» вопросов преподавателя. | Дан недостаточно полный ответ. Студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи. Речевое оформление требует поправок, коррекции. | Не получены ответы по базовым вопросам дисциплины или дан неполный ответ и допущены грубые ошибки. Речь неграмотная. Уточняющие вопросы преподавателя не приводят к коррекции ответа студента не только на поставленный вопрос, но и на другие вопросы дисциплины. |

Пересчет полученной суммы баллов по дисциплине в оценку производится в соответствии с действующим локальным нормативным актом.

**8.2. Примерный вариант 1 рубежной контрольной работы (теста) (15 баллов)**

1.Решить в целых числах уравнение (ax + b) : (cx + d) = 0. (5б)

Входные данные

Вводятся 4 числа: a, b, c и d; c и d не равны нулю одновременно.

Выходные данные

Необходимо вывести все целочисленные решения, если их число конечно, “NO” (без кавычек), если целочисленных решений нет, и “INF” (без кавычек), если их бесконечно много.

2.Дано натуральное число A > 1. Определите, каким по счету числом Фибоначчи оно является, то есть выведите такое число n, что φn=A. Если А не является числом (5б) Фибоначчи, выведите число -1.

Входные данные

Вводится натуральное число.

Выходные данные

Выведите ответ на задачу.

3.Дано натуральное число N. Напишите функцию int NumberOfZeroes(int n) (C#), определяющую количество нулей среди всех цифр числа N. (5б)

Входные данные

Задано единственное число N

Выходные данные

Необходимо вывести количество нулей среди всех цифр числа N.

**8.3. Примерный вариант 2 рубежной контрольной работы (теста) (15 баллов)**

1.Дана строка, состоящая из строчных латинских букв и пробелов. Проверьте, является ли она палиндромом без учета пробелов (например, "аргентина манит негра"). (5б)

Входные данные

На вход подается 1 строка длины не более 100, содержащая пробелы. Подряд может идти произвольное число пробелов.

Выходные данные

Необходимо вывести yes, если данная строка является палиндромом, и no в противном случае.

2.Даны три числа (a, b, c). Поменяйте значения (строго в указанном порядке): a и b, c и a, b и c (5б)

Входные данные

Вводятся три числа 0 ≤ a, b, c ≤ 1000

Выходные данные

Выведите числа поменяв их значения. (См. пример).

Примечание

Нельзя выводить c,b,a. Нужно менять значения!

3.Дан массив, состоящий из целых чисел. Напишите программу, которая подсчитает количество элементов массива, больших предыдущего (элемента с предыдущим номером). (5б)

Входные данные

Сначала задано число NN — количество элементов в массиве (1≤N≤100001≤N≤10000). Далее через пробел записаны NN чисел — элементы массива. Массив состоит из целых чисел.

Выходные данные

Необходимо вывести единственное число - количество элементов массива, больших предыдущего.

**8.4. Вопросы к зачету**

***1 семестр***

1. Операторы ввода/вывода. Консольный ввод и вывод строк и чисел
2. Основные базовые и структурированные типы данных, их характеристика
3. Условные переходы
4. Циклические переходы
5. Основные понятия программирования.
6. Алгоритм. Основные свойства алгоритма. Алгоритмизация задач.
7. Типовые структуры алгоритма.
8. Основные элементы языка. Основы С# (идентификаторы, переменные, выражения,

операции, операторы, атрибуты, спецификаторы).

1. Типы данных (значимые, ссылочные, преобразование, упаковка / распаковка).
2. Стандартные и пользовательские типы данных (символьные, логические, целые,

вещественные, структуры, перечисления, массивы, строки).

1. Функции и методы. Способы передачи данных.
2. Массивы одномерные. Типовые алгоритмы обработки массивов.
3. Массивы как объекты.
4. Строковый тип данных
5. Двумерные массивы.
6. Ступенчатые (“зубчатые”) массивы.
7. Типовые алгоритмы обработки двумерных массивов.

***2 семестр***

1. Файлы. Типы файлов.
2. Создание, открытие, вывод из файла, запись в файл.
3. Объектно-ориентированная парадигма: понятия объекта, класса объектов; основные понятия объектно-ориентированного программировании (инкапсуляция, наследование и полиморфизм); классы и объекты; интерфейсы и реализация.
4. Делегаты и события
5. ООП и Windows приложения.
6. Настройка формы Windows приложения.
7. Размещение элементов управления на форме.
8. События. Создание обработчика события. Основные события компонентов (перечислить и охарактеризовать).
9. Свойства элементов, динамическое изменение.
10. Кнопки. Переключатели, кнопки множественного выбора, контейнеры.
11. Компонент ListBox, его свойства, доступ к объктам, находящимся в списке.
12. Создание многооконных приложений.
13. Элемент управления dataGridView.
14. Компонент отображения графической информации Сhart. Построение графиков и диаграмм.
15. Жизненный цикл программного обеспечения.
16. Типы приложений.
17. Структуры. Стандартные списки, очереди.
18. Стеки Stack

***3 семестр***

1. Технология создания программного кода.
2. Технологии коллективной разработки программного обеспечения.
3. Технологические средства разработки программного обеспечения.
4. Методы отладки и тестирования программ.
5. Категории программных ошибок.
6. Основы анализа эффективности алгоритмов.
7. Асимптотические обозначения и как их применяют для анализа эффективности алгоритмов.
8. Системный стек. Анализ сложности рекурсивных алгоритмов.
9. Сортировка. Основные задачи сортировки. Устойчивая и неустойчивая сортировки.
10. Алгоритм сортировки слиянием.
11. Алгоритма пузырьковой сортировки и ее временная сложность
12. Алгоритм бинарного поиска.
13. Метод быстрой сортировки.
14. Алгоритм «Поиск в глубину»
15. Алгоритм «Поиска в ширину»

***4 семестр***

1. Определение операторов.
2. Перегрузка операций преобразования типов.
3. Индексаторы.
4. Pattern matching. Паттерн типов и свойств, паттерны кортежей, позиционный паттерн, реляционный и логический паттерны, паттерны списков
5. Многопоточность.
6. Класс Thread. Создание потоков.
7. Делегат ThreadStart.
8. Потоки с параметрами и ParameterizedThreadStart.
9. Синхронизация потоков.
10. Мониторы. Класс AutoResetEvent.
11. Мьютексы. Семафоры
12. LINQ: Проекция данных. Фильтрация коллекции.
13. LINQ: Сортировка.
14. LINQ: Объединение, пересечение и разность коллекций.
15. LINQ: Агрегатные операции. Получение части коллекции.
16. LINQ: Группировка. Соединение коллекций.
17. Проверка наличия и получение элементов. Отложенное и немедленное выполнение LINQ.
18. Сборка мусора, управление памятью и указатели. Сборщик мусора в C#.
19. Финализируемые объекты.
20. Метод Dispose. Конструкция using. Указатели. Указатели на структуры, члены классов и массивы.
21. Сериализация в JSON. JsonSerializer
22. Работа с XML в C#. XML-Документы.
23. Работа с XML с помощью классов System.Xml. Изменение XML-документа. XPath. Linq to Xml.
24. Создание Xml-документа. Выборка элементов в LINQ to XML.
25. Изменение XML-документа в LINQ to XML.

**Практическая часть**

1. Прямая на плоскости может быть задана уравнением ax + by = c, где a, b, c – целые; a и b одновременно не равны нулю. Пусть даны коэффициенты нескольких прямых a1, b1, c1, a2, b2, c2, … , an, bn, cn. Определить, имеются ли среди этих прямых прямые, пересекающиеся в одной точке.
2. Даны две последовательности по n чисел в каждой. Найти наименьшее среди тех чисел первой последовательности, которые не входят во вторую (считать, что есть хотя бы одно такое число).
3. Даны натуральное число n, целые числа a, x1, x2, … , xn. Если в последовательности x1, x2, … , xn есть хотя бы один член, равный a , то получить сумму всех членов, следующих за первым таким членом, иначе найти минимальный среди нечетных чисел последовательности x1, x2, … , xn.
4. Даны целые числа a1, a2,…, an, среди которых могут быть повторяющиеся. Составить новый массив из чисел, которые входят в последовательность по одному разу.
5. Даны целые числа a1, a2,…, an, среди которых могут быть повторяющиеся. Составить новый массив из чисел, взятых по одному из каждой группы равных членов данной последовательности.
6. Даны натуральные числа k, n , действительные числа a1, a2,…, an. Получить последовательность max(a1,a2,…,ak), max(ak+1,ak+2,…,a2k),…, max(ak(n-1)+1,…,akn).
7. Дана строка, ограниченная точкой и состоящая из латинских букв, в которой слова разделены одним или несколькими пробелами. Напечатать те слова строки, которые отличны от последнего слова и удовлетворяют следующему свойству: в слове гласные буквы (a, e, i, o, u) чередуются с согласными.
8. Дана строка, ограниченная точкой, в которой слова разделены одним или несколькими пробелами. Напечатать все слова, отличные от последнего слова, предварительно преобразовав каждое из них по следующему правилу: перенести первую букву в конец слова.
9. Дана строка, ограниченная точкой, в которой слова разделены одним или несколькими пробелами. Напечатать все слова, отличные от последнего слова, предварительно преобразовав каждое из них по следующему правилу: перенести последнюю букву в начало слова.
10. Дан двумерный массив. Вставить нулевую строку и нулевой столбец перед строкой и столбцом, где находится первый минимальный элемент.
11. Удалить все строки и столбцы, на пересечении которых стоят отрицательные элементы.
12. Дан массив целых чисел. Вставить число k после всех элементов, кратных своему номеру.

**9. Учебно-методическое и информационное обеспечение дисциплины**

**а) основная литература:**

1. Кувшинов, Д.Р. Основы программирования : учебное пособие для вузов / Д. Р. Кувшинов. — Москва : Издательство Юрайт, 2020. — 104 с. — (Высшее образование). — ISBN 978-5-534-07559-5. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454667>
2. Огнева, М. В.  Программирование на языке С++: практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2020. — 335 с. — (Высшее образование). — ISBN 978-5-534-05123-0. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454165>.
3. Трофимов, В. В.  Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2020. — 137 с. — (Высшее образование). — ISBN 978-5-534-07834-3. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452333>.

**б) дополнительная литература:**

1. Довбуш, Галина Visual C++ на примерах / Галина Довбуш , Анатолий Хомоненко. - М.: БХВ-Петербург, 2012. - 528 c.
2. Пахомов, Б. C/C++ и MS Visual C++ 2010 для начинающих / Б. Пахомов. - М.: БХВ-Петербург, 2011. - 736 c
3. Пахомов, Борис C/C++ и MS Visual C++ 2012 для начинающих / Борис Пахомов. - Москва: СИНТЕГ, 2015. - 518 c.
4. Поляков, А. Методы и алгоритмы компьютерной графики в примерах на Visual C++ / А. Поляков, В. Брусенцев. - М.: БХВ-Петербург, 2011. - 560 c.
5. Понамарев, В. Программирование на C++/C# в Visual Studio .NET 2003 / В. Понамарев. - М.: БХВ-Петербург, 2015. - 917 c.
6. Роберт, С. Сикорд Безопасное программирование на C и C++ / Роберт С. Сикорд. - Москва: РГГУ, 2014. - 496 c.
7. Сидорина, Татьяна Самоучитель Microsoft Visual Studio C++ и MFC / Татьяна Сидорина. - М.: "БХВ-Петербург", 2014. - 848 c.

**в) Профессиональные базы данных и другие интернет-ресурсы:**

***–*** Энциклопедия языков программирования (<http://progopedia.ru>);

– Библиотека программиста ([https://proglib.io](https://proglib.io/));

– Сайт для IT-специалистов (<https://habr.com/ru/all/>);

– Курсы программирования (платные и бесплатные): <https://stepik.org/catalog>, <https://practicum.yandex.ru/catalog/programming/>;

– ЭБС «Научная электронная библиотека eLibrary.ru» (<http://www.elibrary.ru>);

– ЭБС «Университетская библиотека onLine» (<http://www.biblioclub.ru>);

– ЭБС «Юрайт» (<http://www.urait.ru/>);

– Универсальная база данных «ИВИС» ([htpps:/eivis.ru/](about:blank));

– ИС «Национальнаяэлектронная библиотека (НЭБ)»(<https://rusneb.ru/>).

**10. Материально-техническое обеспечение дисциплины**

Для проведения лабораторных занятий используются:

– учебные аудитории для проведения лекционных занятий, занятий семинарского типа, текущего контроля, промежуточной аттестации, оборудованные аудиторной мебелью, доской (меловой, маркерной или интерактивной), компьютером или ноутбуком с возможностью подключения к сети «Интернет» и доступом в электронную информационно-образовательную среду СОГУ, мультимедийным проектором, экраном;

– компьютерный класс (корпус 10, ауд. №505, 506, 600, 601, 605, 606), оборудованный аудиторной мебелью, доской (меловой, маркерной или интерактивной), компьютерами или ноутбуками с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде СОГУ, мультимедийным проектором, экраном.

*Лицензионное программное обеспечение:*

1. Windows 10 Pro for Workstations, (№ 4100072800 Microsoft Products (MPSA) от 04.2016г);
2. Office Standard 2016 (№ 4100072800 Microsoft Products (MPSA) от 04.2016г);
3. Система поиска текстовых заимствований «Антиплагиат ВУЗ».

*Перечень ПО в свободном доступе:* Kaspersky Free; WinRar; Google Chrome; Yandex Browser; Opera Browser; Acrobat Reader; Visual Studio, С++.

Помещение для самостоятельной работы студентов: Зал электронных ресурсов Научной библиотеки СОГУ (корпус 6, кабинет № 1.8), укомплектован специализированной мебелью (рабочие места студентов), необходимыми техническими средствами обучения: компьютеры, принтер, возможность подключения к сети «Интернет», доступ в электронную информационно-образовательную среду СОГУ.