

Документ подписан простой электронной подписью
Информация о владельце:

ФИО: Исаев Игорь Магомедович

Должность: Проректор по учебной работе

Дата подписания: 28.08.2023 14:46:38

Уникальный идентификатор документа:

d7a26b9e8ca85e98ec3de2eb454b4659d061f249

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное образовательное учреждение
высшего образования**

«Национальный исследовательский технологический университет «МИСИС»

Рабочая программа дисциплины (модуля)

Программирование и алгоритмизация

Закреплена за подразделением

Кафедра инфокоммуникационных технологий

Направление подготовки

09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Профиль

Квалификация

Бакалавр

Форма обучения

очная

Общая трудоемкость

5 ЗЕТ

Часов по учебному плану

180

Формы контроля в семестрах:

в том числе:

экзамен 1

аудиторные занятия

68

самостоятельная работа

58

часов на контроль

54

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	1 (1.1)		Итого	
	УП	РП	УП	РП
Неделя	18			
Вид занятий	УП	РП	УП	РП
Лекции	34	34	34	34
Лабораторные	34	34	34	34
Итого ауд.	68	68	68	68
Контактная работа	68	68	68	68
Сам. работа	58	58	58	58
Часы на контроль	54	54	54	54
Итого	180	180	180	180

Программу составил(и):

к.т.н., доц., Стучилин В.В.

Рабочая программа

Программирование и алгоритмизация

Разработана в соответствии с ОС ВО:

Самостоятельно устанавливаемый образовательный стандарт высшего образования - бакалавриат Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский технологический университет «МИСиС» по направлению подготовки 09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ (приказ от 02.04.2021 г. № 119 о.в.)

Составлена на основании учебного плана:

09.03.02 Информационные системы и технологии, 09.03.02-БИСТ-23.plx , утвержденного Ученым советом НИТУ МИСИС в составе соответствующей ОПОП ВО 22.06.2023, протокол № 5-23

Утверждена в составе ОПОП ВО:

09.03.02 Информационные системы и технологии, , утвержденной Ученым советом НИТУ МИСИС 22.06.2023, протокол № 5-23

Рабочая программа одобрена на заседании

Кафедра инфокоммуникационных технологий

Протокол от 12.04.2023 г., №9

Руководитель подразделения Кузнецова Ксения Александровна

1. ЦЕЛИ ОСВОЕНИЯ

1.1	Изучение методов с средств программирования для овладения знаниями в области разработки приложений, на основе современных методов программирования и алгоритмизации; подготовка студентов к осознанному использованию, как методов, так и средств программирования.
-----	---

2. МЕСТО В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Блок ОП:		Б1.О
2.1	Требования к предварительной подготовке обучающегося:	
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:	
2.2.1	Объектно-ориентированное программирование	
2.2.2	Основы дискретной математики	
2.2.3	Комбинаторика и теория графов	
2.2.4	Технологии программирования	
2.2.5	Алгоритмы дискретной математики	
2.2.6	Операционные системы и среды	
2.2.7	Разработка клиент-серверных приложений	
2.2.8	Сетевые технологии	
2.2.9	Теория алгоритмов	
2.2.10	Язык программирования Python	
2.2.11	Методы оптимизации	
2.2.12	Производственная практика по получению профессиональных умений и опыта профессиональной деятельности	
2.2.13	Производственная практика по получению профессиональных умений и опыта профессиональной деятельности	
2.2.14	Технологии разработки ПО	
2.2.15	Нормы и правила оформления НИР и ВКР	
2.2.16	Проектирование информационных систем	
2.2.17	Инфокоммуникационные системы и сети	
2.2.18	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.19	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.20	Управление разработкой по методологии Agile	

3. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ФОРМИРУЕМЫМИ КОМПЕТЕНЦИЯМИ

ОПК-6: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
Знать:
ОПК-6-31 Методы проектирования программного обеспечения с использованием Си#.
ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
Знать:
ОПК-2-31 Основные понятия и методы разработки современного программного обеспечения.
ОПК-6: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
Уметь:
ОПК-6-У1 Разрабатывать алгоритмическое и программное обеспечение для информационных систем.
ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
Уметь:
ОПК-2-У1 Применять полученные теоретические знания при решении практических задач разработки приложений.

ОПК-6: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
Владеть:
ОПК-6-В1 Иметь навыки разработки эффективных приложений для информационных систем
ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
Владеть:
ОПК-2-В1 Иметь навыки использования современных программных средств анализа и тестирования программного обеспечения.

4. СТРУКТУРА И СОДЕРЖАНИЕ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Формируемые индикаторы компетенций	Литература и эл. ресурсы	Примечание	КМ	Выполняемые работы
	Раздел 1. Основные понятия программирования и алгоритмизации							
1.1	Понятие алгоритма. Понятие о структурном подходе к разработке алгоритмов. Метод пошаговой детализации. Основные понятия программирования. /Лек/	1	4	ОПК-2-31 ОПК-6-31	Л1.1Л2.1 Л2.2 Э1 Э2			
1.2	Проработка лекционного материала в электронной системе обучения Canvas Самостоятельное изучение литературы. Подготовка к лабораторным работам. Работа над рефератом. /Ср/	1	2	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2			
	Раздел 2. Основные конструкции Си#							
2.1	Переменные. Типы данных. Литералы. Операторы объявления. Инициализация переменных. /Лек/	1	4	ОПК-2-31 ОПК-6-31	Л1.1Л2.1 Л2.2 Э1 Э2			
2.2	Работа с одномерными, двумерными, ступенчатыми массивами. Обработка исключений. /Лек/	1	4	ОПК-2-31 ОПК-6-31	Л1.1Л2.1 Л2.2 Э1 Э2			
2.3	Ввод–вывод данных. /Лек/	1	4	ОПК-2-31 ОПК-6-31	Л1.1Л2.1 Л2.2 Э1 Э2			
2.4	Разработка программ циклической структуры /Лаб/	1	4	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ1	Р1
2.5	Организация разветвлений. Разветвления в цикле. /Лаб/	1	4	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ2	Р2
2.6	Массивы. Типовые алгоритмы обработки массивов. /Лаб/	1	4	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ3	Р3
2.7	Матрицы. Типовые алгоритмы обработки матриц. /Лаб/	1	4	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ4	Р4

2.8	Проработка лекционного материала в электронной системе обучения Canvas Самостоятельное изучение литературы. Подготовка к лабораторным работам. Работа над рефератом. /Ср/	1	26	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2			
2.9	Математические операторы. Логические операторы. Циклы и ветвления. /Лек/	1	4	ОПК-6-31 ОПК-2-31	Л2.2 Э2			
2.10	Работа с файлами и строками. /Лек/	1	4	ОПК-6-31 ОПК-2-31	Л2.1 Э1			
Раздел 3. Введение в ООП								
3.1	Объектно-ориентированная парадигма в программировании. /Лек/	1	4	ОПК-2-31 ОПК-6-31	Л1.1Л2.1 Л2.2 Э1 Э2			
3.2	Работа со структурами. /Лаб/	1	6	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ5	Р5
3.3	Работа с классами. /Лаб/	1	6	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ6	Р6
3.4	Использование методов. /Лаб/	1	6	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2		КМ7	Р7
3.5	Проработка лекционного материала в электронной системе обучения Canvas Самостоятельное изучение литературы. Подготовка к лабораторным работам. Работа над рефератом. /Ср/	1	30	ОПК-2-У1 ОПК-2-В1 ОПК-6-У1 ОПК-6-В1	Л1.1Л2.1 Л2.2 Э1 Э2			Р8
3.6	Классы и структуры. /Лек/	1	4	ОПК-6-31 ОПК-2-31	Л1.1 Э2			
3.7	Члены класса. /Лек/	1	2	ОПК-6-31 ОПК-2-31	Л2.2 Э1			

5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ

5.1. Контрольные мероприятия (контрольная работа, тест, коллоквиум, экзамен и т.п), вопросы для самостоятельной подготовки

Код КМ	Контрольное мероприятие	Проверяемые индикаторы компетенций	Вопросы для подготовки
КМ1	Защита результатов работы №1.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	1. Что такое цикл? Операторы цикла for, while, do ... while. Различия между ними. 2. Какие данные необходимы для организации цикла for? Что такое управляющая переменная цикла? 3. Циклы по условию и их организация. 4. Операторы break и continue. В каких случаях они используются? 5. Типовые алгоритмы циклической структуры: вычисление суммы n слагаемых, вычисление произведения n множителей, вычисление факториала, табулирование функции. 6. Вычисление суммы с использованием рекуррентных соотношений. 7. Вложенные циклы.

КМ2	Защита результатов работы №2.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	<ol style="list-style-type: none"> 1. Что такое разветвление? Какой оператор используется для организации разветвлений? 2. Обход и его реализация на языке C#. 3. Множественный выбор и его реализация на языке C#. 4. Логические переменные, логические операторы, логические выражения и их использование в программах. 5. Организация обработки потока данных, количество которых заранее не известно. Использование специального значения. 6. Особенности организации ввода, если каждая порция данных включает несколько значений. 7. Что такое множественный выбор? Как он реализуется?
КМ3	Защита результатов работы №3.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	<ol style="list-style-type: none"> 1. Понятие массива. Элемент массива. Индекс элемента массива. Описание массива. Ввод массива. Организация вывода массива в строку, в столбец. 2. Суммирование элементов массива. Суммирование элементов массива, удовлетворяющих условию. 3. Формирование другого массива из элементов заданного массива, удовлетворяющих условию. Перестановка элементов массива. 4. Удаление элемента массива. Включение элемента в массив. 5. Нахождение максимального (минимального) элемента массива. 6. Упорядочение элементов массива. 7. Массивы как объекты. Доступ к методам и свойствам экземпляра типа Array. 8. Доступ к методам типа Array.
КМ4	Защита результатов работы №4.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	<ol style="list-style-type: none"> 1. Матрица. Описание матрицы. Ввод матрицы по строкам, по столбцам. Вывод матрицы. 2. Суммирование элементов матрицы. Суммирование элементов матрицы, удовлетворяющих условию. 3. Нахождение максимального (минимального) элемента матрицы. 4. Суммирование элементов строк (столбцов) матрицы с формированием одномерного массива. 5. Операции со строками (столбцами) матрицы (поиск максимального элемента, включение, удаление элемента, перестановка элементов и т.п.) 6. Удаление строки (столбца) матрицы. Включение одномерного массива в качестве строки (столбца) в матрицу. 7. Операции с главной диагональю, с побочной диагональю. 8. Обработка фрагмента матрицы (верхнего, нижнего треугольника; верхней, нижней, правой, левой четверти; периметра и т.п.). 9. Особенности обработки матрицы, заданной в виде одномерной последовательности. 10. Вывод по строкам матрицы, заданной в виде одномерной
КМ5	Защита результатов работы №5.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	<ol style="list-style-type: none"> 1. Структура. Поля структуры. Члены структуры. Объявление структуры. Доступ к отдельным полям структуры. 2. Создание экземпляра структуры. Инициализация полей структуры. 3. Операции со структурами. 4. Преимущества использования структур. 5. Массивы структур и их обработка. 6. Создание объекта структуры при помощи конструктора. Что такое конструктор экземпляра. 7. Особенности инициализации полей структуры при использовании конструктора экземпляра. 8. Различные способы задания значений полей структуры при использовании конструктора. 9. Возможность использования методов, как членов структуры.

КМ6	Защита результатов работы №6.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	<ol style="list-style-type: none"> 1. Что такое класс? К какому типу относится класс? Как определить класс? 2. Каковы основные члены класса? 3. Как создать экземпляр класса? 4. Инициализация полей класса. 5. Объявление массива классов. Особенности выделения памяти под массив классов. 6. Использование конструктора экземпляра при работе с классами. 7. Что такое наследование классов и как оно реализуется? 8. Укажите основные различия между классами и структурами.
КМ7	Защита результатов работы №7.	ОПК-2-У1;ОПК-2-В1;ОПК-6-У1;ОПК-6-В1	<ol style="list-style-type: none"> 1. Что такое метод. Разновидности методов: метод, возвращающий значение, и метод, не возвращающий значения. Особенности их оформления. 2. Что такое подпись метода. 3. Вызов метода. Способы передачи параметров: по значению, по ссылке. Правила согласования формальных и фактических параметров при вызове метода. 4. Различные возможности взаимного расположения вызываемого и вызывающего методов: в одном классе, в разных классах. 5. Особенности вызова нестатического метода из статического метода. 6. Использование массивов в качестве параметров. 7. Методы как параметры: Использование делегата для передачи метода как параметра другого метода.
КМ8	Реферат	ОПК-2-31;ОПК-6-31;ОПК-6-У1;ОПК-2-У1	<p>Примеры тем рефератов:</p> <ol style="list-style-type: none"> 1. Делегаты в C#. 2. События в C#. 3. Анонимные методы в C#. 4. Лямбды в C#. 5. Ковариантность и контравариантность делегатов 6. Делегаты Action, Predicate и Func в C#. 7. Встроенные методы (Expression-Bodied) в C#. 8. Необобщенные коллекции в C#. 9. Обобщенные коллекции в C#. 10. Сериализация в C#. 11. Сборщик мусора в C#. 12. Класс ObservableCollection. 13. Индексаторы и создание коллекций. 14. Интерфейсы IEnumerable и IEnumerator. 15. Итераторы и оператор yield. 16. Работа с дисками. 17. Работа с каталогами. 18. Работа с файлами. Классы File и FileInfo. 19. Чтение и запись файла. Класс FileStream. 20. Чтение и запись текстовых файлов. StreamReader и StreamWriter.

КМ9	Контрольная работа №1.	ОПК-2-31;ОПК-6-31	<ol style="list-style-type: none"> 1. Что такое алгоритм? Блок-схемы. Примеры. 2. Зачем нужны переменные? Примеры. 3. Для чего и как используются комментарии в программе на Си#? Примеры. 4. Для чего служат операторы выбора? Примеры. 5. Как выглядит стандартная запись цикла for? Примеры. 6. Механизм работы .NET Framework. 7. Какие преимущества компиляторов перед интерпретаторами? Примеры. 8. Базовые типы переменных языка Си#. Примеры. 9. Что такое предупреждающие сообщения компилятора? Примеры. 10. Как выглядит стандартная запись оператора if? Примеры. 11. Приведите пример программы с использованием оператора while. 12. Основные назначения оператора перехода break? Примеры. 13. Что такое алгоритм? Исполнитель алгоритма. Пример исполнителя алгоритма. 14. Операторы консольного ввода/вывода. Примеры. 15. Какие требования предъявляются к именам переменных? Примеры. 16. Что такое оператор строгого равенства? Примеры. 17. Что такое бесконечный цикл? Как его реализовать? Примеры. 18. Основные назначения оператора перехода continue? Примеры. 19. Для чего служит компилятор? Примеры. 20. Что такое синтаксические ошибки в программе? Примеры.
КМ10	Контрольная работа №2.	ОПК-2-31;ОПК-6-31	<ol style="list-style-type: none"> 1. Определение массива. Примеры использования. 2. Объявление одномерного массива. Примеры. 3. Варианты инициализации массива. Примеры. 4. Объявление многомерного массива. Примеры. 5. Ступенчатые (jagged) массивы в Си#. Примеры. 6. Обработка исключений в Си#. Примеры. 7. Типы исключений. Примеры обработки. 8. Обработка исключения FileNotFoundException. Пример. 9. Обработка исключения DivideByZeroException. Пример. 10. Назначение блока finally в обработчике исключения. Пример. 11. Работа с файлами. Класс File. Примеры. 12. Работа с файлами. Класс FileStream. Примеры. 13. Режимы открытия файла при использовании класса FileStream . Примеры. 14. Чтение данных из файла. Примеры. 15. Запись данных в файл. Примеры. 16. Работа с директориями в Си#. Примеры. 17. Работа со строками в Си#. Примеры. 18. Методы класса String для работы со строками в Си#. Примеры. 19. Свойства класса String для работы со строками в Си#. Примеры. 20. Использование специальных символов при работе со строками. Примеры.

KM11	Экзамен.	ОПК-2-31;ОПК-6-31	<ol style="list-style-type: none"> 1. Переменные в Си#. Типы данных. Примеры. 2. Арифметические операции в Си#. Примеры. 3. Преобразование типов данных. Явное и неявное преобразование. Примеры 4. Логические операции в Си#. Примеры. 5. Ключевое слово var. Примеры. 6. Оператор if-else. Примеры. 7. Оператор switch. Примеры. 8. Тернарный оператор «?:». Примеры. 9. Цикл for. Примеры. 10. Цикл while. Примеры. 11. Цикл do/while. Примеры. 12. Цикл foreach. Примеры. 13. Оператор try-catch. Примеры. 14. Операторы break и continue . Примеры. 15. Типы исключений. Примеры. 16. Объявление одномерного массива. Примеры. 17. Обработка исключения DivideByZeroException. Пример. 18. Работа с файлами. Класс FileStream. Примеры. 19. Методы класса String для работы со строками в Си#. Примеры. 20. Классы в Си#. Примеры. 21. Методы в Си#. Примеры. 22. Назначение блока finally в обработчике исключения. Пример. 23. Запись данных в файл. Примеры. 24. Работа со строками в Си#. Примеры. 25. Основные принципы объектно-ориентированного программирования. Примеры. 26. Работа со структурами в Си#. Примеры. 27. Варианты инициализации массива. Примеры. 28. Обработка исключения FileNotFoundException. Пример. 29. Режимы открытия файла при использовании класса FileStream . Примеры. 30. Работа с директориями в Си#. Примеры. 31. Создание объектов в Си#. Примеры. 32. Конструкторы в Си#. Примеры. 33. Объявление многомерного массива. Примеры. 34. Обработка исключений в Си#. Примеры. 35. Работа с файлами. Класс File. Примеры. 36. Свойства класса String для работы со строками в Си#. Примеры. 37. Парадигма объектно-ориентированного программирования. Примеры. 38. Константы в Си#. Примеры. 39. Ступенчатые (jagged) массивы в Си#. Примеры. 40. Типы исключений. Примеры обработки. 41. Чтение данных из файла. Примеры. 42. Использование специальных символов при работе со строками. Примеры. 43. Поля класса в Си#. Примеры. 44. Отличия классов от структур в Си#. Примеры. 45. Класс List. Примеры использования. 46. Правила оформления блок-схем. Блок условия. Примеры. 47. Правила оформления блок-схем. Блок цикла. Примеры. 48. Правила оформления блок-схем. Оформление методов. Примеры. 49. Передача аргументов в методы. Аргументы по значению. Примеры. 50. Передача аргументов в методы. Аргументы по ссылке.
------	----------	-------------------	---

5.2. Перечень работ, выполняемых по дисциплине (Курсовая работа, Курсовой проект, РГР, Реферат, ЛР, ПР и т.п.)

Код работы	Название работы	Проверяемые индикаторы компетенций	Содержание работы
------------	-----------------	------------------------------------	-------------------

P1	Лабораторная работа №1. Разработка программ циклической структуры.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Для предлагаемой задачи определить, что является ее решением. Определить тип переменных. Выделить повторяющиеся действия и записать их в общем виде. Определить условие окончания цикла и вид цикла. Составить программу. Проверить ее работу вручную и на компьютере.</p> <ol style="list-style-type: none"> 1. Определить наибольшее значение множителя n, для которого произведение $p = 1 \cdot 4 \cdot 7 \cdot \dots \cdot n$ не превышает $L = 30\,000$. 2. Определить частное и остаток от деления двух целых чисел N и M, используя операцию вычитания. 3. Начав тренировки, спортсмен в первый день пробежал 10 км. Каждый следующий день он увеличивал дневную норму на 10 % от нормы предыдущего дня. Определить: <ol style="list-style-type: none"> а) какой суммарный путь пробежит спортсмен за 7 дней; б) через сколько дней спортсмен пробежит суммарный путь 100 км; в) через сколько дней спортсмен будет пробегать в день больше 20 км?
P2	Лабораторная работа №2. Организация разветвлений. Разветвления в цикле.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Задание требует сочетания циклов и разветвлений, а также использования множественного выбора. Предполагается, что количество вводимых исходных данных n задано.</p> <ol style="list-style-type: none"> 1. Определить средний рост девочек и мальчиков одного класса. В классе учатся n учеников. 2. В компьютер вводятся по очереди координаты n точек. Определить, сколько из них попадет в круг радиусом r с центром в точке (a, b). 3. Ученику 1-го класса назначается дополнительно стакан молока (200 мл), если его вес составляет меньше 30 кг. Определить, сколько литров молока потребуется ежедневно для одного класса, состоящего из n учеников. После взвешивания вес каждого ученика вводится в компьютер.
P3	Лабораторная работа №3. Массивы. Типовые алгоритмы обработки массивов.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Программу составить так, чтобы она могла быть использована для обработки массивов произвольного размера. В программе предусмотреть печать исходных данных и печать результата с поясняющим текстом. Подготовить тесты и проверить работу программы на компьютере.</p> <ol style="list-style-type: none"> 1. Минимальный элемент заданного одномерного массива увеличить в два раза. 2. В одномерном массиве найти сумму элементов, расположенных до максимального элемента массива. 3. Все элементы одномерного массива, расположенные перед минимальным, увеличить в 2 раза.
P4	Лабораторная работа №4. Матрицы. Типовые алгоритмы обработки матриц.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Программу составить так, чтобы она могла быть использована для обработки массивов произвольного размера. В программе предусмотреть печать исходных данных и печать результата с поясняющим текстом. Подготовить тесты и проверить работу программы на компьютере.</p> <ol style="list-style-type: none"> 1. Найти сумму элементов матрицы A размера 5×7. 2. Найти среднее среди положительных элементов матрицы A размером 5×7. 3. Найти след (сумму диагональных элементов) квадратной матрицы A размера 4×4. 4. Определить номер строки и столбца, содержащих минимальный элемент матрицы A размера 3×6.

P5	Лабораторная работа №5. Работа со структурами.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Решить задачи с использованием конструктора и различных вариантов объявления структуры (внутри класса, вне класса). Данные задать самостоятельно. Составить программу и проверить ее решение на компьютере.</p> <p>1. Результаты соревнований по прыжкам в длину определяются по сумме двух попыток. В протоколе для каждого участника указываются: фамилия, общество, результаты первой и второй попыток. Вывести протокол в виде таблицы с заголовком в порядке занятых мест.</p> <p>2. Составить программу для обработки результатов кросса на 500 м для женщин. Для каждой участницы ввести фамилию, группу, фамилию преподавателя, результат. Получить результирующую таблицу, упорядоченную по результатам, в которой содержится также информация о выполнении норматива. Определить суммарное количество участниц, выполнивших норматив.</p> <p>3. Радиокomпания провела опрос слушателей по вопросу: «Кого вы считаете человеком года?». Определить пять наиболее часто встречающихся ответов и их долей (в процентах от общего количества ответов).</p>
P6	Лабораторная работа №6. Работа с классами.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Выполнить задание к ЛР №5 с использованием классов, используя наследование по указанию преподавателя.</p>
P7	Лабораторная работа №7. Использование методов.	ОПК-2-У1;ОПК-2-В1	<p>Пример задания: Определить параметры методов. Разработать методы. Составить программу. Проверить ее работу на компьютере.</p> <p>1. Поменять местами максимальные элементы матриц А размера 5×6 и В размера 3×5. Поиск максимального элемента матрицы осуществить в методе.</p> <p>2. В массивах А размера 9 и В размера 7 заменить максимальные элементы на среднее арифметическое значение элементов, расположенных после максимального, в том массиве, для которого максимальный элемент расположен дальше от конца массива. Поиск максимального элемента осуществить в методе.</p>
P8	Реферат	ОПК-2-31;ОПК-6-31	<p>Необходимо написать реферат на выбранную студентом тему. Реферат должен быть оформлен по шаблону, должен содержать ссылки на источники. Должно быть использовано не менее трех источников.</p>
P9	Контрольная работа №1.	ОПК-2-31;ОПК-6-31	<p>1. Основные понятия алгоритмизации и программирования. 2. Типы переменных. 3. Объявление и инициализация переменных. 4. Операторы в Си#. 5. Циклы и ветвления.</p>
P10	Контрольная работа №2.	ОПК-2-31;ОПК-6-31	<p>1. Одномерные массивы. 2. Матрицы. 3. Ввод-вывод в Си#. 4. работа с файлами м строками. 5. Основные понятия ООП.</p>

5.3. Оценочные материалы, используемые для экзамена (описание билетов, тестов и т.п.)

Экзамен проводится в письменной форме. Билет состоит из 3 теоретических вопросов и двух задач.

Оценочные материалы (средства) представляют собой задания, процедуры защиты отчетов: а) по выполненной лабораторных работ; б) по представленному реферату; в) по результатам проверки контрольных работ.

Указанные оценочные средства позволяют оценить компетенции, сформированные у обучающихся в процессе освоения дисциплины.

Оформленный реферат, отчет по домашней и лабораторным работам в электронной форме загружаются обучающимися в ЭИОС «Canvas».

Экзаменационные вопросы:

1. Переменные в Си#. Типы данных. Примеры.
2. Арифметические операции в Си#. Примеры.
3. Преобразование типов данных. Явное и неявное преобразование. Примеры
4. Логические операции в Си#. Примеры.
5. Ключевое слово var. Примеры.
6. Оператор if-else. Примеры.
7. Оператор switch. Примеры.
8. Тернарный оператор «?:». Примеры.
9. Цикл for. Примеры.
10. Цикл while. Примеры.
11. Цикл do/while. Примеры.
12. Цикл foreach. Примеры.
13. Оператор try-catch. Примеры.
14. Операторы break и continue . Примеры.
15. Типы исключений. Примеры.
16. Объявление одномерного массива. Примеры.
17. Обработка исключения DivideByZeroException. Пример.
18. Работа с файлами. Класс FileStream. Примеры.
19. Методы класса String для работы со строками в Си#. Примеры.
20. Классы в Си#. Примеры.
21. Методы в Си#. Примеры.
22. Назначение блока finally в обработчике исключения. Пример.
23. Запись данных в файл. Примеры.
24. Работа со строками в Си#. Примеры.
25. Основные принципы объектно-ориентированного программирования. Примеры.
26. Работа со структурами в Си#. Примеры.
27. Варианты инициализации массива. Примеры.
28. Обработка исключения FileNotFoundException. Пример.
29. Режимы открытия файла при использовании класса FileStream . Примеры.
30. Работа с директориями в Си#. Примеры.
31. Создание объектов в Си#. Примеры.
32. Конструкторы в Си#. Примеры.
33. Объявление многомерного массива. Примеры.
34. Обработка исключений в Си#. Примеры.
35. Работа с файлами. Класс File. Примеры.
36. Свойства класса String для работы со строками в Си#. Примеры.
37. Парадигма объектно-ориентированного программирования. Примеры.
38. Константы в Си#. Примеры.
39. Ступенчатые (jagged) массивы в Си#. Примеры.
40. Типы исключений. Примеры обработки.
41. Чтение данных из файла. Примеры.
42. Использование специальных символов при работе со строками. Примеры.
43. Поля класса в Си#. Примеры.
44. Отличия классов от структур в Си#. Примеры.
45. Класс List. Примеры использования.
46. Правила оформления блок-схем. Блок условия. Примеры.
47. Правила оформления блок-схем. Блок цикла. Примеры.
48. Правила оформления блок-схем. Оформление методов. Примеры.
49. Передача аргументов в методы. Аргументы по значению. Примеры.
50. Передача аргументов в методы. Аргументы по ссылке. Примеры.

5.4. Методика оценки освоения дисциплины (модуля, практики. НИР)

По дисциплине предусмотрен письменный экзамен. Билет состоит из трех теоретических вопросов и двух типовых задач.

Экзаменационная оценка:

Оценка "отлично" выставляется студенту, полностью ответившему на три теоретических вопроса и решившему две задачи экзаменационного билета, обнаружившему всестороннее, систематическое и глубокое знание учебного материала, предусмотренного программой; усвоившему основную и знакомому с дополнительной литературой по программе; умеющему творчески и осознанно выполнять задания, предусмотренные программой; усвоившему взаимосвязь основных понятий и умеющему применять их к анализу и решению практических задач; безусловно выполнившему в процессе изучения дисциплины все задания, предусмотренные формами текущего контроля;

Оценки "хорошо" заслуживает студент, ответивший полностью на два вопроса и решившему одну задачу экзаменационного билета и ответивший частично на другой вопрос, при этом обнаруживший полное знание учебного материала, предусмотренного программой; успешно выполнивший все задания, предусмотренные формами текущего контроля;

Оценка "удовлетворительно" выставляется студенту, ответившему полностью только на один вопрос и решившему одну задачу экзаменационного билета или допустившему погрешности в ответе на вопросы экзаменационного билета и обладающему необходимыми знаниями для их устранения под руководством преподавателя;

Оценка "неудовлетворительно" выставляется студенту, не ответившему на три теоретических вопроса, или ответившему на один вопрос и не решившему ни одной задачи экзаменационного билета, обнаружившему пробелы в знании основного материала, предусмотренного программой, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий; не выполнившему отдельные задания, предусмотренные формами текущего контроля.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ**6.1. Рекомендуемая литература****6.1.2. Дополнительная литература**

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л2.1	Сорокин А. А.	Объектно-ориентированное программирование: учебное пособие: курс лекций	Электронная библиотека	Ставрополь: Северо-Кавказский Федеральный университет (СКФУ), 2014
Л2.2	Мурадханов С. Э., Широков А. И.	Информатика и программирование. Объектно-ориентированное программирование (на основе языка C): учебник	Электронная библиотека	М.: Изд-во МИСиС, 2015
Л2.3	Мурадханов С. Э., Широков А. И.	Информатика и программирование. Основы разработки программ на языке C: учебник	Электронная библиотека	М.: Изд-во МИСиС, 2013

6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Э1	Codenet	http://www.codenet.ru/
Э2	Документация по Си#	https://docs.microsoft.com/ru-ru/dotnet/csharp/

6.3 Перечень программного обеспечения

П.1	Microsoft Visio 2016
П.2	Microsoft Visual Studio 2015
П.3	Microsoft SQL server 2016
П.4	Microsoft Office
П.5	LMS Canvas
П.6	MS Teams

6.4. Перечень информационных справочных систем и профессиональных баз данных

И.1	Полнотекстовые российские научные журналы и статьи:
И.2	— Научная электронная библиотека eLIBRARY https://elibrary.ru/
И.3	Иностранные базы данных (доступ с IP адресов МИСиС):
И.4	— аналитическая база (индексы цитирования) Web of Science https://apps.webofknowledge.com
И.5	— аналитическая база (индексы цитирования) Scopus https://www.scopus.com/

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ		
Ауд.	Назначение	Оснащение
Л-731	Учебная аудитория	доска аудиторная меловая, экран проекционный, проектор, стационарные компьютеры 15 шт. ПО-Visual Studio; Electronic WorkBench; APACHE; MySQL; XAMPP; Python, комплект учебной мебели, пакет лицензионных программ MS Office
Л-728	Учебная аудитория	доска аудиторная меловая, экран проекционный, проектор, стационарные компьютеры 15 шт. ПО-Visual Studio; Electronic WorkBench; APACHE; MySQL; XAMPP; Python; комплект учебной мебели, пакет лицензионных программ MS Office
Б-934	Лекционная аудитория	4 кабины для синхронного перевода, мультимедийные экраны и проектор, ноутбук, пакет лицензионных программ MS Office, комплект учебной мебели на 130 посадочных мест

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

Подготовка к лекциям.

Подготовка к лекционному занятию включает выполнение всех видов заданий, рекомендованных к каждой лекции, т.е. задания выполняются еще до лекционного занятия по соответствующей теме.

В ходе лекционных занятий необходимо вести конспектирование учебного материала, обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. При необходимости задавать преподавателю уточняющие вопросы. Работая над конспектом лекций, необходимо использовать основную и дополнительную литературу, которую рекомендовал лектор.

Подготовка к практическим или лабораторным занятиям

Подготовка к каждому практическому или лабораторному занятию должна начинаться с ознакомления с планом практического или лабораторного занятия, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованной к данной теме.

В процессе подготовки к практическим или лабораторным занятиям, необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. При всей полноте конспектирования лекции в ней невозможно изложить весь материал из-за лимита аудиторных часов. Поэтому самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала.

Самостоятельная работа по теоретическому курсу включает аудиторную и внеаудиторную самостоятельную работу на лекциях, работу с лекционным материалом после лекции, выполнение дополнительных индивидуальных заданий на практических занятиях и лабораторных работах.