

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Аксенова Татьяна Алексеевна

Должность: Директор

Дата подписания: 25.06.2021 09:15

Идентификатор ключа:

6f9e8fef93cabde10122c8f7fc53725f900c0bb6ec4d7b290b531dcdaadce5ea

Профессиональная образовательная организация ассоциация
«Региональный финансово-экономический техникум»

Цикловая комиссия математических, естественнонаучных и экономических дисциплин



Рабочая программа учебной дисциплины

«ПРОГРАММИРОВАНИЕ»

специальности **09.02.05 Прикладная информатика (по отраслям)**
(базовой подготовки)

Курск 2020

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования (далее – СПО) 09.02.05 «Прикладная информатика (по отраслям)» (базовой подготовки), утвержденного приказом Министерства образования и науки Российской Федерации от 13 августа 2014 года № 1001.

Составитель:



А.С. Смецкой, преподаватель
цикловой комиссии математических,
естественнонаучных и
экономических дисциплин РФЭТ

Рабочая программа утверждена на заседании цикловой комиссии математических, естественнонаучных и экономических дисциплин, протокол № 10 от «26» июня 2020 г.

Председатель цикловой комиссии математических,
естественнонаучных и экономических дисциплин

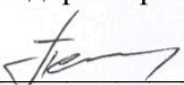


В.Н. Бутова

**Изменения в рабочей программе по учебной дисциплине
«Программирование»
на 2021/2022 уч. г.**

УТВЕРЖДАЮ

Зам. директора по учебной работе


 Ю.И. Петренко

«25» июня 2021 г.

В рабочую программу вносятся следующие изменения:

- 1) внесены изменения в перечень вопросов для самоконтроля по самостоятельно изученным темам;
- 2) внесены изменения в перечень вопросов для подготовки к экзамену.

Рабочая программа утверждена на заседании цикловой комиссии математических, естественно-научных и экономических дисциплин, протокол № 8 от «25» июня 2021 г.

Председатель цикловой комиссии  В.Н. Бутова

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ7
3. КОМПЛЕКТ КОНТРОЛЬНО–ОЦЕНОЧНЫХ СРЕДСТВ12
4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ13
5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ15

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ «Программирование»

1.1. Область применения программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности СПО 09.02.05 Прикладная информатика (по отраслям) (базовой подготовки) в части освоения профессионального цикла.

1.2. Место дисциплины в структуре программы подготовки специалистов среднего звена: дисциплина является вариативной в профессиональном цикле.

1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины:

Целью освоения дисциплины «Программирование» является приобретение теоретических знаний и практических навыков в освоении языка программирования C#; формирование теоретических основ функционального программирования (лямбда-исчисление, комбинаторная логика) и практических навыков применения функционального подхода; знакомство с некоторыми теоретическими разделами компьютерных наук.

Задачи изучения дисциплины:

- сформировать у студента представление о структуре языка, его месте среди языков программирования, области задач, для которых он наиболее подходит;
- познакомить с элементами языка, философией подготовки и обработки данных;
- привить практические навыки программирования на C#: использование самые современных возможностей программирования, умение создавать правильный и легко читаемый код;
- показать направления использования преимуществ высокопроизводительных вычислительных платформ.

В результате изучения дисциплины обучающийся должен:
уметь:

- ориентироваться в современных тенденциях развития языков программирования (У-1);
- применять интегрированную среду разработки Visual Studio (У-2);
- анализировать готовый код на предмет синтаксических ошибок, вносить необходимые изменения (У-3);
- работать с основными элементами языка C#, типами данных, операторами, массивами и строками (У-4);

- обрабатывать разнотипные данные с соблюдением типовой безопасности (У-5);
- управлять средствами ввода-вывода, обработки исключительных ситуаций (У-6);

знать:

- особенности современных языков программирования: С, С#, Java; их достоинства и ограничения; круг задач, для решения которых они наиболее подходят (З-1);
- основные элементы языка С# (З-2);
- типы данных, литералы, переменные изучаемого языка программирования, методы работы с ними (З-3);
- классификацию операторов и порядок работы с ними в языке С# (З-4);
- особенности работы с массивами и строками в языке С# (З-5);
- порядок работы с классами, объектами и методами (З-6);
- принципы работы программ, использующих наследование (З-7);
- основы обработки исключительных ситуаций (З-8);
- организацию системы ввода-вывода в С# на потоках (З-9)

1.4. Перечень формируемых компетенций в результате освоения учебной дисциплины

Код	Наименование результата обучения
ОК 1	Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес
ОК 2	Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество
ОК 3	Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность
ОК 4	Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития
ОК 5	Использовать информационно-коммуникационные технологии в профессиональной деятельности.
ОК 8	Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.
ОК 9	Ориентироваться в условиях частой смены технологий в профессиональной деятельности.
ПК 2.2.	Разрабатывать и публиковать программное обеспечение и информационные ресурсы отраслевой направленности со статическим и динамическим контентом на основе готовых спецификаций и стандартов.
ПК 2.3.	Проводить отладку и тестирование программного обеспечения.
ПК 2.4.	Проводить адаптацию отраслевого программного обеспечения.

1.5. Рекомендуемое количество часов на освоение примерной региональной программы учебной дисциплины:

максимальной учебной нагрузки обучающегося 260 часов, в том числе:

обязательной аудиторной учебной нагрузки обучающегося 76 часов;

самостоятельной работы обучающегося 184 часа.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	260
Обязательная аудиторная учебная нагрузка (всего)	76
в том числе:	
3 курс	
обзорно-установочные занятия	14
лабораторно-практические занятия	20
контрольная работа	+
4 курс	
обзорно-установочные занятия	2
лабораторно-практические занятия	20
контрольная работа	+
Самостоятельная работа обучающегося (всего)	184
Итоговая аттестация проводится по курсам в форме дифференцированного зачета/экзамена (соответственно)	

2.2. Тематический план и содержание учебной дисциплины «Программирование»

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, контрольные работы	Объем часов	Уровень освоения
Тема 1. Введение в предмет. Создание C#	Содержание учебного материала Современные языки и направления программирования. Генеалогическое дерево C#. Язык C – начало современной эпохи программирования. Появление объектно-ориентированного программирования (ООП) и C#. Появление Интернета и Java. Создание и развитие C#. Связь C# со средой .NET Framework. Принцип действия CLR. Управляемый и неуправляемый код. Общезыковая специализация.	2	1,2
	Практическое занятие 1 Среда .NET Framework. Действие CLR. Псевдокод MSIL	4	
	Самостоятельная работа 1. Проработка учебного материала. 2. Исследование тенденций развития современных языков программирования.	18	
Тема 2. Основные элементы языка	Содержание учебного материала Сущность объектно-ориентированного программирования. Инкапсуляция Полиморфизм. Наследование. Первая простая программа. Применение компилятора командной строки csc.exe. Применение интегрированной среды разработки Visual Studio. Построчный анализ первого примера программы. Обработка синтаксических ошибок. Незначительное изменение программы. Другие типы данных. Два управляющих оператора. Условный оператор. Оператор цикла. Использование кодовых блоков. Точка с запятой и оформление исходного текста программы. Ключевые слова C#. Идентификаторы. Библиотека классов среды .NET Framework	4	1,2
	Практическое занятие 2 Программа с созданием двух переменных.	4	
	Самостоятельная работа: 1. Проработка учебного материала. 2. Подготовка к тестированию.	18	
Тема 3. Типы данных, литералы и переменные	Содержание учебного материала Значение типа данных в языке C#. Типы значений в C#. Целочисленные типы. Типы для представления чисел с плавающей точкой. Десятичный тип данных. Символы. Логический тип данных. Некоторые возможности вывода. Литералы. Шестнадцатеричные литералы. Управляющие последовательности символов. Строковые литералы. Инициализация переменной. Динамическая инициализация. Неявно типизированные переменные. Область действия и время существования переменных. Преобразование и приведение типов. Автоматическое преобразование типов. Приведение несовместимых типов. Преобразование типов в выражениях. Приведение типов в выражениях.	2	2
	Практическое занятие 3 Детальная работа с переменными (bool, int, char)	4	

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, контрольные работы	Объем часов	Уровень освоения	
	Самостоятельная работа: <ol style="list-style-type: none"> 1. Проработка учебного материала. 2. Подготовка к тестированию 3. Анализ программ с применением типов в приложениях 	18		
Тема 4. Операторы. Управляющие операторы	Содержание учебного материала Операторы. Арифметические операторы. Операторы инкремента и декремента. Операторы отношения и логические операторы. Укороченные логические операторы. Оператор присваивания. Составные операторы присваивания. Поразрядные операторы. Поразрядные операторы И, ИЛИ, исключающее ИЛИ и НЕ. Операторы сдвига. Поразрядные составные операторы присваивания. Оператор ?. Использование пробелов и круглых скобок. Предшествование операторов Управляющие операторы. Оператор if. Вложенные операторы if. Конструкция if-else-if. Оператор switch. Вложенные операторы switch. Оператор цикла for. Некоторые разновидности оператора цикла for. Применение нескольких переменных управления циклом. Условное выражение. Отсутствующие части цикла. Бесконечный цикл. Циклы без тела. Объявление управляющих переменных в цикле for. Оператор цикла while. Оператор цикла do-while. Оператор цикла foreach. Применение оператора break для выхода из цикла. Применение оператора continue. Оператор return. Оператор goto	4	3	
	Практическое занятие 4 Анализ примеров кода с операторами	4		
	Самостоятельная работа: <ol style="list-style-type: none"> 1. Проработка учебного материала. 2. Подготовка к тестированию по вопросам темы. 3. Классификация операторов 	18		
Тема 5. Введение в классы, объекты и методы	Содержание учебного материала Основные положения о классах. Общая форма определения класса. Определение класса Создание объектов. Переменные ссылочного типа и присваивание. Методы. Добавление метода в класс Building. Возврат из метода. Возврат значения. Использование параметров. Добавление параметризованного метода в класс Building. Исключение недоступного кода. Конструкторы. Параметризованные конструкторы. Добавление конструктора в класс Building. Оператор new. «Сборка мусора» и применение деструкторов. Ключевое слово this	2	3	
	Практическое занятие 5 Действия деструкторов.	4		
	Самостоятельная работа: <ol style="list-style-type: none"> 1. Проработка учебного материала 2. Подготовка к тестированию по теме. 3. Подготовка к дифференцированному зачету 	20		
Тема 6. Массивы и строки	Содержание учебного материала		3	

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, контрольные работы	Объем часов	Уровень освоения
	<p>Массивы. Одномерные массивы. Инициализация массива. Соблюдение границ массива. Многомерные массивы. Двумерные массивы. Массивы трех и более измерений. Инициализация многомерных массивов. Ступенчатые массивы. Присваивание ссылок на массивы. Применение свойства Length. Применение свойства Length при обращении со ступенчатыми массивами. Неявно типизированные массивы. Оператор цикла foreach. Строки. Построение строк. Обращение со строками. Массивы строк. Постоянство строк. Применение строк в операторах switch</p> <p>Практическое занятие 6 Работа с массивами</p> <p>Самостоятельная работа:</p> <ol style="list-style-type: none"> 1. Проработка учебного материала. 2. Подготовка к тестированию по теме 	8	
<p>Тема 7. Классы, объекты и методы</p>	<p>Содержание учебного материала Управление доступом к членам класса. Модификаторы доступа. Организация закрытого и открытого доступа. Практический пример организации управления доступом. Передача объектов методам по ссылке. Способы передачи аргументов методу. Использование модификаторов параметров ref и out. Использование модификатора параметра ref. Использование модификатора параметра out. Использование модификаторов ref и out для ссылок на объекты. Использование переменного числа аргументов. Возврат объектов из методов. Перегрузка методов. Перегрузка конструкторов. Инициализаторы объектов. Необязательные аргументы. Именованные аргументы. Метод Main(). Рекурсия. Ключевое слово static. Применение ключевого слова static. Статические конструкторы. Статические классы</p> <p>Практическое занятие 7 Создание простого кода по заданной ситуации.</p> <p>Самостоятельная работа:</p> <ol style="list-style-type: none"> 1. Проработка учебного материала. 2. Подготовка к тестированию по теме 	2	3
<p>Тема 8. Наследование</p>	<p>Содержание учебного материала Основы наследования. Доступ к членам класса и наследование. Организация защищенного доступа. Конструкторы и наследование. Вызов конструкторов базового класса. Наследование и сокрытие имен. Создание многоуровневой иерархии классов. Порядок вызова конструкторов. Ссылки на базовый класс и объекты производных классов. Виртуальные методы и их переопределение. Применение виртуальных методов. Применение абстрактных классов. Предотвращение наследования с помощью ключевого слова sealed. Класс object. Упаковка и распаковка. Класс object как универсальный тип данных</p>		3

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, контрольные работы	Объем часов	Уровень освоения
	Практическое занятие 8 Создание и анализ простого кода по заданным параметрам	8	
	Самостоятельная работа 1. Проработка учебного материала. 2. Подготовка к тестированию по теме	18	
Тема 9. Обработка исключительных ситуаций	Содержание учебного материала Основы обработки исключительных ситуаций. Последствия перехвата исключений. Обработка исключительных ситуаций как способ устранения программных ошибок. Применение нескольких операторов catch. Вложение блоков try. Генерирование исключений вручную. Использование блока finally. Подробное рассмотрение класса Exception. Получение производных классов исключений. Перехват исключений производных классов. Применение ключевых слов checked и unchecked		3
	Практическое занятие 9 Создание и анализ простого кода по заданным параметрам	8	
	Самостоятельная работа 1. Проработка учебного материала. 2. Подготовка к тестированию по теме	18	
Тема 10. Применение средств ввода-вывода	Содержание учебного материала Организация системы ввода-вывода в C# на потоках. Классы потоков. Консольный ввод-вывод. Класс FileStream и байтовый ввод-вывод в файл. Открытие и закрытие файла. Чтение байтов из потока файлового ввода-вывода. Запись в файл. Использование класса Filestream для копирования файла. Символьный ввод-вывод в файл. Переадресация стандартных потоков. Чтение и запись двоичных данных. Классы потоков BinaryReader и BinaryWriter. Демонстрирование двоичного ввода-вывода		3
	Практическое занятие 10 Создание и анализ простого кода по заданным параметрам	8	
	Самостоятельная работа: 1. Проработка учебного материала. 2. Анализ особенностей современных поисковых систем, их популярности и статистики распределения. 3. Подготовка к экзамену.	18	
	Итого	260	

3. КОМПЛЕКТ КОНТРОЛЬНО–ОЦЕНОЧНЫХ СРЕДСТВ

Смотри приложение №1.

4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ

4.1. Требования к минимальному материально-техническому обеспечению

1. Аудиторная база (лекционная аудитория, аудитория для проведения практических занятий, виртуальные классные комнаты на портале РФЭТ)
2. Организационно-технические средства и аудиовизуальный фондовый материал, мультимедийное оборудование.
3. Комплекты видеofilьмов, аудиокниг, CD-дисков по проблемам дисциплины.
4. Интернет.

4.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Основы программирования: учебник [Электронный ресурс]; Региональный финансово-экономический техникум. — Курск, 2015. — 257 с.
2. Основы программирования: практикум [Электронный ресурс]; Региональный финансово-экономический техникум. — Курск, 2015. — 56 с.
3. Дунаев В. Web-программирование для всех. — Санкт-Петербург: БХВ-Петербург 2012 г.— 560 с. — Электронное издание. — ISBN 978-5-9775-0197-2

Дополнительная

1. Герберт Шилдт Полное руководство C# 4.0.- ИД «Вильямс», 2013.- 1056с.
2. Эндрю Троелсен Язык программирования C# 2010 и платформа .NET 4 - ИД «Вильямс», 2014.-1392с.
3. Новиков Ф.А. Дискретная математика (для программистов). М.: Питер. 2000.
4. Бочаров В.А., Маркин В.И. Основы логики: Учебник. – М.: ИД «Форум»: ИНФРА-М, 2008. – 336 с.
5. Горский Д.П. Краткий словарь по логике. М.,1991.
6. Попов, Ю.П. Логика: учеб. пособие / 3-е изд., перераб. и доп. М.: КНОРУС, 2009. 304 с.
7. Хаггарти Р. Дискретная математика для программистов. Москва: Техносфера, 2012. – 400 с., ISBN 978-5-94836-303-5

Интернет-ресурсы:

1. Учебный курс для студентов специальности «Прикладная информатика» в системе электронного обучения eduterra.pro

2. Электронная библиотека Регионального финансово-экономического техникума
<http://students.rfet.ru/a/students/library.jspx>
3. Электронная библиотека Регионального финансово-экономического института /
<http://lib2.rfei.ru/>
4. Электронно - библиотечная система iBooks
<http://ibooks.ru/>
5. Федеральный портал «Российское образование»
<http://www.edu.ru/>
6. Федеральное хранилище «Единая коллекция цифровых образовательных ресурсов»
<http://school-collection.edu.ru/>
7. Российская Государственная Библиотека
<http://www.rsl.ru/>
8. Фундаментальная библиотека СПбГПУ –
<http://www.unilib.neva.ru/rus/lib/>

5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий, тестирования, а также выполнения обучающимися индивидуальных заданий.

Результаты обучения (освоенные умения, усвоенные знания, полученные компетенции)	Формы и методы контроля и оценки результатов обучения
1	2
Умения:	<p>Наблюдение и оценка выполнения практических работ</p> <p>Оценка выполнения тестовых заданий</p> <p>Оценка выполнения контрольной работы</p> <p>Оценка выполнения самостоятельной работы</p> <p>Дифференцированный зачет</p> <p>Экзамен</p>
ориентироваться в современных тенденциях развития языков программирования (У-1);	
применять интегрированную среду разработки Visual Studio (У-2);	
анализировать готовый код на предмет синтаксических ошибок, вносить необходимые изменения (У-3);	
работать с основными элементами языка С#, типами данных, операторами, массивами и строками (У-4);	
обрабатывать разнотипные данные с соблюдением типовой безопасности (У-5);	
управлять средствами ввода-вывода, обработки исключительных ситуаций (У-6);	
Знания:	
особенности современных языков программирования: С, С#, Java; их достоинства и ограничения; круг задач, для решения которых они наиболее подходят (З-1);	
основные элементы языка С# (З-2);	
типы данных, литералы, переменные изучаемого языка программирования, методы работы с ними (З-3);	
классификацию операторов и порядок работы с ними в языке С# (З-4);	
особенности работы с массивами и строками в языке С# (З-5);	
порядок работы с классами, объектами и методами (З-6);	
принципы работы программ, использующих наследование (З-7);	
основы обработки исключительных ситуаций (З-8);	
организацию системы ввода-вывода в С# на потоках (З-9)	