

ГОСУДАРСТВЕННОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ТУЛЬСКОЙ ОБЛАСТИ
«ТУЛЬСКИЙ ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ»

УТВЕРЖДАЮ
Директор ГПОУТО «ТЭК»
А.В. Макарова
Приказ № _____
«27» августа 2021 года



**РАБОЧАЯ ПРОГРАММА
УЧЕБНОЙ ДИСЦИПЛИНЫ**

**ОП.4 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ
ПРОГРАММА ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА ПО
СПЕЦИАЛЬНОСТИ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ**

Квалификация СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ

Щекино
2021год

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (далее – ФГОС) по специальности среднего профессионального образования (далее СПО) **09.02.07 Информационные системы и программирование**, утвержденного приказом Министерства образования и науки Российской Федерации 9 декабря 2016 года № 1547.

Организация-разработчик: **Государственное профессиональное образовательное учреждение Тульской области «Тульский экономический колледж»**

Разработчики:

Каргина Ольга Ивановна, преподаватель высшей квалификационной категории государственного профессионального образовательного учреждения Тульской области «Тульский экономический колледж»

Рабочая программа рассмотрена и одобрена предметно-цикловой комиссией № 3 Государственного профессионального образовательного учреждения Тульской области «Тульский экономический колледж»

Утверждена протоколом № 1 от «27» августа 2021 года

Председатель ПЦК №3 _____ П.Е.Панюхина

Заместитель директора по учебной работе _____ Е.В.Кошелева

«27» августа 2021 года

СОДЕРЖАНИЕ

	стр.
1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	7
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ	16
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ	18

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.4 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС СПО **09.02.07 Информационные системы и программирование**, входящей в укрупненную группу специальностей **09.00.00 Информатика и вычислительная техника**.

1.2. Место дисциплины в структуре программы подготовки специалистов среднего звена:

Учебная дисциплина «Основы алгоритмизации и программирования» принадлежит к профессиональному циклу (П.00) и базируется на знаниях основных математических операций и связи с дисциплиной «Элементы высшей математики».

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

В результате освоения дисциплины обучающийся должен уметь:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.
- Выполнять проверку, отладку кода программы

В результате освоения дисциплины обучающийся должен знать:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм.
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения.

В результате освоения учебной дисциплины у студента должны формироваться общие и профессиональные компетенции, включающие в себя способность:

Код	Наименование общих компетенций
ОК 1.	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.
ОК 2.	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 4.	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
ОК 5.	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
ОК 6	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
ОК 9.	Использовать информационные технологии в профессиональной деятельности.
ОК 10.	Пользоваться профессиональной документацией на государственном и иностранном языках.
ОК 11	Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере

Код	Наименование профессиональных компетенций
ПК 1.1.	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
ПК 1.2.	Разрабатывать программные модули в соответствии с техническим заданием.
ПК 1.3.	Выполнять отладку программных модулей с использованием специализированных программных средств.
ПК 1.4.	Выполнять тестирование программных модулей.
ПК 1.5.	Осуществлять рефакторинг и оптимизацию программного кода.
ПК 1.6.	Разрабатывать модули программного обеспечения для мобильных платформ.
ПК 2.4.	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.
ПК 2.5.	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.
ПК 3.1.	Осуществлять ревьюирование программного кода в соответствии с технической документацией.
ПК 3.3.	Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма.
ПК 3.4.	Проводить сравнительный анализ программных продуктов и средств разработки, с целью выявления наилучшего решения согласно критериям, определенным техническим заданием.

1.4. Количество часов на освоение программы дисциплины:

максимальной учебной нагрузки студента **292** часа, в том числе:
обязательной аудиторной учебной нагрузки студента **272** часа;
самостоятельной работы студента **20** часов.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Суммарная учебная нагрузка во взаимодействии с преподавателем	272
Самостоятельная работа	20
Объем образовательной программы	292
в том числе:	
теоретическое обучение	136
практические занятия	136
самостоятельная работа студента	20
в том числе:	
Контрольная работа по теме «Основные понятия алгоритмизации» (по вариантам)	2
Решение задач по образцу по теме «Операции и выражения»	2
Решение задач по образцу по теме «Управляющие структуры»	2
Решение задач по образцу по теме «Операторы цикла»	2
Контрольная работа по теме «Структуры данных» (по вариантам)	2
Решение задач по образцу по теме «Потоки. Файлы»	2
Подготовка реферата с презентацией по теме «Визуальные компоненты»	2
Подготовка реферата по теме «Способы реализации интерфейсов»	2
Решение задач по образцу по теме «Структуры и перечисления»	2
Решение задач по образцу по теме «Коллекции. Делегаты. События и потоки выполнения»	2
Промежуточная аттестация проводится в форме экзамена в 3 семестре и дифференцированного зачета 4 семестре	

2.2 Тематический план и содержание учебной дисциплины ОП.4 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Уровень освоения	Объем часов	Осваиваемые элементы компетенций
1	2	3	4	5
Раздел 1. Основы алгоритмизации			28	
Тема 1.1. Основные понятия алгоритмизации	Содержание учебного материала		10	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1-ПК 1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4
	1 Понятие алгоритма, его свойства и виды. Критерии «хорошего» алгоритма. Способы описания алгоритмов: псевдокоды, блок-схема, программа.	1		
	2 Базовые алгоритмические конструкции: линейная, разветвляющаяся, циклическая. Блок-схема: основные элементы, правила составления	2		
	3 Различные комбинации алгоритмических конструкций. Тестовые данные. Математическая модель алгоритма.	2		
	4 Классические алгоритмы Евклида	2		
	5 Алгоритмы решения нелинейных и линейных уравнений	2		
	Практические занятия		16	
	1 Практическая работа №1. Составление блок-схем линейных алгоритмов	3		
	2 Практическая работа №2. Составление блок-схем алгоритмов ветвления	3		
	3 Практическая работа №3. Составление блок-схем циклических алгоритмов	3		
	4 Практическая работа №4. Составление блок-схем алгоритмов усложненной структуры	3		
	5 Практическая работа №5. Составление блок-схем алгоритмов поиска минимального элемента	3		
	6 Практическая работа №6. Составление блок-схем алгоритмов бинарного поиска	3		
	7 Практическая работа №7. Составление блок-схем алгоритмов сортировки массива простым выбором	3		
	8 Практическая работа №8. Составление блок-схем алгоритмов сортировки массива методом обмена	3		
	Самостоятельная работа обучающихся контрольная работа по теме «Основные понятия алгоритмизации»		2	
Раздел 2. Основы программирования			148	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1-ПК 1.6, ПК 2.4, 2.5,
Тема 2.1.	Содержание учебного материала		20	

Основные элементы процедурного языка	1	История языков программирования. Генеалогическое древо C#. Преимущества и недостатки языка C#.	1	ПК 3.1, 3.3, 3.4
	2	Платформа .NET Framework: назначение, составные элементы, тонкости компиляции. Обзор интегрированной среды разработки VisualStudio .NET.	2	
	3	Создание, сохранение и закрытие проектов и решений. Структура программы. Точка входа.	2	
	4	Базовый синтаксис C#. Пространство имен. Классы.	2	
	5	Типы данных: значимые и ссылочные. Преобразование типов: явное и неявное.	2	
	6	Переменные: определение, правила именования. Объявление переменных и их инициализация. Область действия и время существования переменных.	2	
	7	Константы: определение, виды и правила записи в программе. Суффиксы целых и вещественных констант. Escape – последовательности.	2	
	8	Понятие выражения. Математические операторы. Старшинство операторов.	2	
	9	Математические функции (класс Math).	2	
	10	Ввод – вывод данных. Операторы присваивания.	2	
Тема 2.2. Управляющие структуры	Лабораторные работы		10	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1-ПК 1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4
	1	Лабораторная работа №1. Знакомство со средой Visual.Studio.Net	3	
	2	Лабораторная работа №2. Создание консольного приложения	3	
	3	Лабораторная работа №3. Организация простейшего ввода-вывода	3	
	4	Лабораторная работа №4. Разработка линейных программ. Операции и выражения	3	
	5	Лабораторная работа №5. Использование математических функций класса Math	3	
	Самостоятельная работа обучающихся		2	
	Решение задач по образцу по теме «Операции и выражения»			
	Содержание учебного материала		12	
	1	Операторы отношения. Оператор if-else: назначение и правила записи. Проверка простых и сложных условий. Вложенные условные операторы.	2	
2	Использование поразрядных (!, &, , ^) и условных (&&,) логических операторов. Отличие в использовании операторов & и &&, и .	2		
3	Тернарный оператор: правила использования.	2		
4	Механизм получения случайного числа.	2		
5	Оператор выбора switch-case: назначение и правила записи.	2		
6	Операторы перехода: break, goto, return.	2		
Лабораторные работы		6		

	1	Лабораторная работа №6. Создание проектов разветвляющейся структуры	3		
	2	Лабораторная работа №7. Создание проектов с использованием логических операторов	3		
	3	Лабораторная работа №8. Создание проектов с использованием оператора выбора switch-case	3		
	Самостоятельная работа обучающихся			2	
	Решение задач по образцу по теме «Управляющие структуры»				
Тема 2.3.	Содержание учебного материала			12	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1-1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4
Операторы цикла	1	Цикл с предусловием (while)	2		
	2	Цикл с постусловием (do)	2		
	3	Цикл с параметром (for).	2		
	4	Получение таблицы значений	2		
	5	Нахождение суммы и произведения элементов ряда.	2		
	6	Принудительный выход из цикла: операторы break и continue. Бесконечные циклы.	2		
	Лабораторные работы			10	
	1	Лабораторная работа №9. Создание проектов с использованием циклического оператора while	3		
	2	Лабораторная работа №10. Создание проектов с использованием циклического оператора do	3		
	3	Лабораторная работа №11. Создание проектов с использованием циклического оператора for	3		
	4	Лабораторная работа №12. Создание проектов с использованием break, goto, return	3		
	5	Лабораторная работа №13. Решение задач с применением циклов	3		
	Самостоятельная работа обучающихся			2	
Решение задач по образцу по теме «Операторы цикла»					
Тема 2.4.	Содержание учебного материала			22	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1-1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4
Структуры данных	1	Массивы: определение, виды. Объявление одномерного массива. Варианты инициализации.	2		
	2	Ввод и вывод одномерных массивов.	2		
	3	Стандартные операции для работы с массивами: заполнение случайными значениями и значениями по формуле	2		
	4	Нахождение суммы и произведения	2		
	5	Нахождение максимума (минимума)	2		

6	Подсчет количества элементов, удовлетворяющих определенному условию.	2
7	Обработка одномерных массивов: сортировка	2
8	Обработка одномерных массивов: поиск элементов	2
9	Цикл foreach.	2
10	Двумерные массивы: объявление, ввод и вывод.	2
11	Работа с двумерными массивами по строкам и по столбцам	2
Лабораторные работы		18
1	Лабораторная работа №14. Создание проектов с использованием одномерных массивов	3
2	Лабораторная работа №15. Создание проектов с использованием двумерных массивов	3
3	Лабораторная работа №16. Создание проектов с использованием методов сортировки	3
4	Лабораторная работа №17. Создание проектов с использованием методов поиска	3
5	Лабораторная работа №18. Создание проектов с использованием цикла foreach	3
6	Лабораторная работа №19. Создание проектов с использованием методов класса Array: Sort, Indexof, BinarySearch	3
7	Лабораторная работа №20. Создание проектов с использованием ступенчатых массивов	3
8	Лабораторная работа №21. Создание проектов с использованием массивов символов	3
9	Лабораторная работа №22. Создание проектов с использованием строк	3
Самостоятельная работа обучающихся Контрольная работа по теме «Структуры данных» (по вариантам)		2
Тема 2.5 Потоки. Файл Содержание учебного материала		12
1	Понятие потока. Механизм буферизации. Классы библиотеки .NET для работы с потоками.	2
2	Виды доступа к файлам. Объект FileStream.	2
3	Классы StreamWriter и StreamReader.	2
4	Основные операции при работе с файлами. Способы работы с текстовыми файлами.	2
5	Обработка текстовых файлов.	2
6	Работа с каталогами и файлами	2
Лабораторные работы		16
1	Лабораторная работа №23. Создание проектов с использованием classRandom	3

ОК 1, ОК 4, ОК 5,
ОК 9, ОК 10, ПК 1.1-
ПК 1.6, ПК 2.4, 2.5,
ПК 3.1, 3.3, 3.4

	2	Лабораторная работа №24. Создание проектов с использованием обработки исключительных ситуаций	3		
	3	Лабораторная работа №25. Создание проектов с использованием операторов checked и unchecked	3		
	4	Лабораторная работа №26. Создание проектов с использованием многопоточных приложений	3		
	5	Лабораторная работа №27. Создание проектов с использованием текстовых файлов	3		
	6	Лабораторная работа №28. Создание проектов с использованием двоичных файлов	3		
	7	Лабораторная работа №29. Создание проектов с использованием асинхронного ввода-вывода	3		
	8	Лабораторная работа №30. Создание проектов с использованием копирования файлов	3		
	Самостоятельная работа обучающихся Решение задач по образцу по теме «Потоки. Файлы»			2	
Раздел 3. Объектно-ориентированная модель программирования				116	
Тема 3.1	Содержание учебного материала			22	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1-ПК 1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4
Основные принципы объектно-ориентированного программирования	1	Понятие класса и объекта. Характеристики объекта: поля, свойства, методы, события.	2		
	2	Основные принципы объектно-ориентированного программирования: наследование, полиморфизм, инкапсуляция.	2		
	3	Общая форма определения класса. Модификаторы доступа к элементам класса: public, private, protected, internal	2		
	4	Примеры создания классов.	3		
	5	Переменные ссылочного типа и присваивание. Побочные эффекты множественных ссылок.	2		
	6	Метод: понятие, правила записи. Вызов метода. Передача параметров по значению. Правило триединого соответствия параметров и аргументов: по количеству, типам и по порядку следования.	2		
	7	Создание методов, возвращающих значения. Способы размещения методов. Конструкторы.	2		
	8	Инкапсуляция как управление доступом к данным. Свойства класса: понятие, виды, правила записи.	2		

9	Наследование и полиморфизм. Иерархия классов: понятие, преимущества. Синтаксис наследования.	2
10	Скрытие и перекрытие методов. Ключевые слова virtual, override. Вызов методов базового класса («родителя»): ключевое слово base.	2
11	Тонкости использования конструкторов в иерархически связанных между собой классах.	2
Лабораторные работы		36
1	Лабораторная работа №31. Решение задач на создание классов со свойствами.	3
2	Лабораторная работа №32. Создание проектов с использованием текстовых элементов управления и окон сообщений	3
3	Лабораторная работа №33. Создание проектов с использованием диалоговых окон	3
4	Лабораторная работа №34. Создание проектов с использованием кнопок	3
5	Лабораторная работа №35. Создание проектов с использованием компонентов для создания меню	3
6	Лабораторная работа №36. Создание проектов с использованием графических объектов	3
7	Лабораторная работа №37. Создание проектов с использованием многооконного интерфейса	3
8	Лабораторная работа №38. Создание проектов с использованием списков	3
9	Лабораторная работа №39. Создание проектов с использованием переключателей	3
10	Лабораторная работа №40. Создание проектов с использованием текстовых редакторов	3
11	Лабораторная работа №41. Создание проектов с использованием перегрузки методов	3
12	Лабораторная работа №42. Создание проектов с использованием рекурсивных методов	3
13	Лабораторная работа №43. Создание проектов с использованием метода Main	3
14	Лабораторная работа №44. Создание проектов с использованием методов с переменным количеством аргументов	3
15	Лабораторная работа №45. Создание классов, иерархически связанных между собой	3
16	Лабораторная работа №46. Создание проектов с использованием абстрактных классов	3
17	Лабораторная работа №47. Создание проектов с использованием виртуальных методов	3

	18	Лабораторная работа №48. Создание проектов с использованием наследования	3			
	Самостоятельная работа обучающихся Подготовка реферата с презентацией по теме «Визуальные компоненты»			2		
Тема 3.2 Интерфейсы	Содержание учебного материала			14	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1- ПК 1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4	
	1	Интерфейсы: назначение, правила написания. Способы реализации интерфейсов.	2			
	2	Работа с объектами через интерфейсы.	2			
	3	Операторы is и as. Восходящее и нисходящее приведение. Наследование в интерфейсах.	2			
	4	Сходства и различия интерфейсов, абстрактных классов и обычных классов.	2			
	5	Стандартные интерфейсы .NET: IComparable	2			
	6	Стандартные интерфейсы .NET: ICloneable	2			
	7	Стандартные интерфейсы .NET: IEnumerable	2			
	Лабораторные работы					10
	1	Лабораторная работа №49. Создание проектов с использованием интерфейса IComparable	3			
2	Лабораторная работа №50. Создание проектов с использованием интерфейса ICloneable	3				
3	Лабораторная работа №51. Создание проектов с использованием интерфейса IEnumerable.	3				
4	Лабораторная работа №52. Создание проектов с использованием операторов is и as	3				
	5	Лабораторная работа №53. Создание проектов с использованием наследования в интерфейсах	3			
	Самостоятельная работа обучающихся Подготовка реферата по теме «Способы реализации интерфейсов»			2		
Тема 3.3 Структуры и перечисления	Содержание учебного материала			4	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1- ПК 1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4	
	1	Назначение и синтаксис структуры. Элементы тела структуры. Сходства и различия структур и классов. Плюсы и минусы использования структур	2			
	2	Тип данных enum: объявление и тонкости использования. Операции с перечислениями.	2			
	Лабораторные работы					4
	1	Лабораторная работа №54. Создание проектов с использованием перечислений	3			
	2	Лабораторная работа №55. Создание проектов с использованием структур	3			

	Самостоятельная работа обучающихся Решение задач по образцу по теме «Структуры и перечисления»		2	
Тема 3.4. Коллекции. Делегаты. События и потоки выполнения	Содержание учебного материала		8	ОК 1, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.1- ПК 1.6, ПК 2.4, 2.5, ПК 3.1, 3.3, 3.4
	1	Понятие коллекции. Класс List: возможности, правила работы, встроенные методы. Работа с ArrayList.	2	
	2	Делегаты: понятие, правила описания. Примеры и тонкости использования делегатов.	2	
	3	Обработка события: автоматическое создание обработчиков.	2	
	4	Создание библиотеки. Использование библиотеки	2	
	Лабораторные работы		10	
	1	Лабораторная работа №56. Решение задач на создание событий.	3	
	2	Лабораторная работа №57. Работа с ArrayList.	3	
	3	Лабораторная работа №58. Создание проектов с использованием делегатов	3	
	4	Лабораторная работа №59. Создание проектов с групповых делегатов	3	
5	Лабораторная работа №60. Создание проектов с использованием обработки события	3		
Самостоятельная работа обучающихся Решение задач по образцу по теме «Коллекции. Делегаты. События и потоки выполнения»		2		
Всего:			292	

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация программы дисциплины требует наличия следующих лабораторий:

1 Лаборатория «Программирования и баз данных»

- Автоматизированные рабочие места на 12-15 обучающихся (Процессор не ниже Core i3, оперативная память объемом не менее 8 Гб);
- Автоматизированное рабочее место преподавателя (Процессор не ниже Core i3, оперативная память объемом не менее 8 Гб);
- Сервер в лаборатории (8-х ядерный процессор с частотой не менее 3 ГГц, оперативная память объемом не менее 16 Гб, жесткие диски общим объемом не менее 1 Тб, программное обеспечение: Windows Server 2012 или более новая) или выделение аналогичного по характеристикам виртуального сервера из общей фермы серверов
- Проектор и экран;
- Маркерная доска;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:
Eclipse IDE for Java EE Developers, .NET Framework SDK 8, Microsoft SQL Server Express Edition, Microsoft Visio Professional, Microsoft Visual Studio, MySQL Installer for Windows, NetBeans, SQL Server Management Studio, Microsoft SQL Server Java Connector, Android Studio, IntelliJ IDEA.

3.2. Информационное обеспечение обучения

Перечень используемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники (печатные издания):

1. Шилдт, Герберт. С# 4.0: полное руководство: Пер. с англ. — М.: ООО "И.Д. Вильямс", 2019. - 1056 с.

Дополнительные источники (печатные издания)

1. Биллиг В.А. Основы программирования на С#. - М.: Изд-во «Интернет-университет информационных технологий - ИНТУИТ .ру», 2006. - 488 с.
2. Вирт Н. Алгоритмы и структуры данных. - СПб.: Невский диалект, 2001. - 352 с.
3. Гуннерсон Э. Введение в С#. Библиотека программиста. - СПб.: Питер, 2001. - 304 с.
4. Дрейер М. С# для школьников: Учебное пособие / М. Дрейер. Перевод с англ. под ред. В. Биллига - М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2009. - 128 с.
5. Петцольд Ч. Программирование для MS Windows на С#. Т. 1. — М.: Издательско-торговый дом «Русская Редакция», 2002. — 576 с.
6. Пономарев В. А. Программирование на С++ / С# в Visual Studio .NET 2003. Серия «Мастер программ». — СПб.: БХВ-Петербург, 2004. — 352 с.
7. Шилдт Г. С#: Учебный курс. - СПб.: Питер, 2002. - 512 с: ил.
8. Microsoft Corporation. Разработка Windows-приложений на Microsoft Visual Basic .NET и Microsoft Visual C# .NET. Учебный курс. Сертификационный экзамен № 70-306, 70-316. — М.: Издательско-торговый дом «Русская Редакция», 2003. - 512 с.
9. Пахомов Б.И. С# для начинающих. Издательство: БХВ-Петербург, 2014. - 432 с.
10. Стиллмен Э., Грин Дж. Изучаем С#. 3-е изд. - СПб.: Питер, 2014. - 816 с.

Интернет-ресурсы

1. С# начинающих. Автор: Пахомов Б. И. Издательство: БХВ-Петербург Год: 2014 Язык: Русский ISBN: 978-5-9775-0943-5 Страниц: 432
Источник: <http://forcoder.ru/c-sharp/pages/2>
2. Изучаем С#. Третье издание Автор: Эндрю Стиллмен, Дженнифер Грин Издательство: 978-5-496-00867-9 Год: 2014 Язык: Русский ISBN: 978-5-496-00867-9 Страниц: 816 Формат: pdf
Размер: 23 Мб
Источник: <http://forcoder.ru/c-sharp/pages/2>
3. Design Patterns via C#. Приемы объектно-ориентированного проектирования

Источник: <http://forcoder.ru/c-sharp/pages/2> Издательство: ITVDN Год: 2015 Язык: Русский
Страниц: 288 Формат: pdf Размер: 13 Мб
Источник: <http://forcoder.ru/c-sharp/pages/2>
4. Оптимизация приложений на платформе .NET
Саха Годштейн, Дима Зурбалева, Идо Флатов Издательство: ДМК Пресс Год: 2014 Язык:
Русский ISBN: 978-5-94074-944-8 Страниц: 522 Формат: pdf
Источник: <http://forcoder.ru/c-sharp/pages/3>

3.3. Организация образовательного процесса

Дисциплина ОП.4 Основы алгоритмизации и программирования изучается в течение 3 и 4 семестров.

Форма проведения консультаций для обучающихся - индивидуальная.

3.4. Кадровое обеспечение образовательного процесса

Реализация образовательной программы обеспечивается педагогическими работниками образовательной организации, а также лицами, привлекаемыми к реализации образовательной программы на условиях гражданско-правового договора, в том числе из числа руководителей и работников организаций, направление деятельности которых соответствует области профессиональной деятельности, имеющих стаж работы в данной профессиональной области не менее 3 лет.

Квалификация педагогических работников образовательной организации должна отвечать квалификационным требованиям, указанным в квалификационных справочниках и профессиональных стандартах.

Педагогические работники, привлекаемые к реализации образовательной программы, должны получать дополнительное профессиональное образование по программам повышения квалификации, в том числе стажировок в организациях направление деятельности, которых соответствует области профессиональной деятельности, не реже 1 раза в 3 года с учетом расширения спектра профессиональных компетенций.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий и лабораторных работ, тестирования, а также выполнения студентами индивидуальных заданий, проектов, исследований.

Результаты обучения (освоенные умения, усвоенные знания)	Основные показатели оценки результата	Формы и методы контроля и оценки
<i>Перечень знаний, осваиваемых в рамках дисциплины:</i>		
Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, алгоритмические конструкции.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	-устный опрос, тестирование -выполнение индивидуальных заданий различной сложности
Эволюция языков программирования, их классификация, понятие системы программирования	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	оценка ответов в ходе эвристической беседы, тестирование
Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	-оценка ответов в ходе эвристической беседы -выполнение индивидуальных заданий различной сложности
Подпрограммы, составление библиотек подпрограмм.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	-оценка ответов в ходе эвристической беседы
Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	-оценка ответов в ходе эвристической беседы -выполнение индивидуальных заданий различной сложности - письменная проверка в виде контрольной работы
<i>Перечень умений, осваиваемых в рамках дисциплины:</i>		
Разрабатывать алгоритмы для конкретных задач	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	-выполнение индивидуальных заданий различной сложности оценка результатов выполнения практических работ

Использовать программы для графического отображения алгоритмов		90-100 % правильных ответов выполненных действий – «5»; 70- 89% правильных ответов выполненных действий – «4»; 50-69 % правильных ответов выполненных действий – «3»; менее 50 % - «2»	и и и	-оценка результатов выполнения практических работ -выполнение индивидуальных заданий различной сложности
Определять сложность работы алгоритмов.		90-100 % правильных ответов выполненных действий – «5»; 70- 89% правильных ответов выполненных действий – «4»; 50-69 % правильных ответов выполненных действий – «3»; менее 50 % - «2»	и и и	-устный опрос -тестирование -оценка результатов выполнения практических работ
Работать в среде программирования.		90-100 % правильных ответов выполненных действий – «5»; 70- 89% правильных ответов выполненных действий – «4»; 50-69 % правильных ответов выполненных действий – «3»; менее 50 % - «2»	и и и	-устный опрос, -тестирование, -оценка результатов выполнения практических работ -выполнение индивидуальных заданий различной сложности
Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.		90-100 % правильных ответов выполненных действий – «5»; 70- 89% правильных ответов выполненных действий – «4»; 50-69 % правильных ответов выполненных действий – «3»; менее 50 % - «2»	и и и	устный опрос, тестирование, -оценка результатов выполнения практических работ -выполнение индивидуальных заданий различной сложности
Оформлять код программы в соответствии со стандартом кодирования		90-100 % правильных ответов выполненных действий – «5»; 70- 89% правильных ответов выполненных действий – «4»; 50-69 % правильных ответов выполненных действий – «3»; менее 50 % - «2»	и и и	-оценка результатов выполнения практических работ
Выполнять проверку, отладку кода программы		90-100 % правильных ответов выполненных действий – «5»; 70- 89% правильных ответов выполненных действий – «4»; 50-69 % правильных ответов выполненных действий – «3»; менее 50 % - «2»	и и и	-оценка результатов выполнения практических работ