

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИ ВлГУ)**

Отделение среднего профессионального образования

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____ Д.Е. Андрианов
« 21 » 05 2024 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

для специальности 09.02.07 Информационные системы и программирование

Муром, 2024 г.

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 №1547 от 09 декабря 2016 года.

Кафедра-разработчик: информационных систем.

Рабочую программу составил: к.т.н., доцент Комкова С.В.

от «07» мая 2024 г.

(подпись)

Рабочая программа рассмотрена и одобрена на заседании кафедры ИС.

Протокол № 18

от «07» мая 2024 г.

Заведующий кафедрой ИС *Андреанов Д.Е.*

(подпись)

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	8
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	9

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

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 по специальности 09.02.07 Информационные системы и программирование.

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании, для получения дополнительных компетенций, умений и знаний, необходимых для обеспечения конкурентоспособности выпускника на рынке труда и продолжения образования по специальности.

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

Дисциплина ОП.04 Основы алгоритмизации и программирования является общепрофессиональной дисциплиной

Дисциплина базируется на школьном курсе информатики и математики, и является базовой для всех специальных дисциплин направления подготовки.

1.3. Цели и задачи учебной дисциплины - требования к результатам освоения учебной дисциплины:

Целью дисциплины является:

- изучение и освоение базовых понятий и приемов программирования, применяемых на всех основных этапах разработки программ;
- изучение методов программирования для овладения знаниями в области технологии программирования;
- подготовка к осознанному использованию как языков программирования, так и методов программирования.

Задачами дисциплины являются:

- формирование целостного представления о предмете;
- освоение теоретических основ структурного программирования;
- изучение конкретных языков программирования;
- использования языков программирования для решения практических задач

В результате освоения дисциплины обучающийся должен знать:

- Демонстрирует знания алгоритмизации, языков и технологий программирования, пригодных для практического применения в области информационных систем и технологий (ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 2.3.).

В результате освоения дисциплины обучающийся должен уметь:

- т применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий (ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 2.3.).

В результате освоения дисциплины обучающийся должен владеть следующими общими (ОК) и профессиональными (ПК) компетенциями:

- 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 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств;

1.4. Количество часов на освоение программы учебной дисциплины:

Максимальной учебной нагрузки обучающегося 59 часов, в том числе:

обязательной аудиторной нагрузки обучающегося 50 часов;

самостоятельной нагрузки обучающегося 9 часов.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
	3 семестр
Максимальная учебная нагрузка (всего)	59
Обязательная аудиторная учебная нагрузка (всего)	50
В том числе:	
лекционные занятия	18
практические занятия	16
лабораторные работы	16
контрольные работы	
курсовая работа	0
Самостоятельная работа обучающегося (всего)	9
Итоговая аттестация в форме	Экзамен

2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
1	2	3	4
	3 семестр		
Раздел 1	Основы алгоритмизации		
Тема 1.1 Принципы решения задач на ЭВМ	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Основные принципы решения задач на ЭВМ. История развития языка C++.	2	1
	<i>Самостоятельная работа обучающихся.</i> Запись алгоритмов в виде структурограмм. Рекомендации по оформлению программ на C++. Пространства имён.	9	3
Раздел 2	Основы C++		
Тема 2.1 Механизмы ввода-вывода	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Синтаксис языка C++. Алфавит и идентификаторы. Типы данных. Объявление переменных. Механизмы ввода и вывода.	4	1
	<i>Практические занятия.</i> Знакомство со средой разработки. Линейные алгоритмы.	2	2
	<i>Лабораторные работы.</i> Математические функции в C++.	4	3
Тема 2.2 Ветвления и циклы	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Операторы выбора. Циклы. Операторы безусловного перехода.	4	1
	<i>Практические занятия.</i> Организация разветвлений. Условный оператор. Операторы выбора. Цикл с параметром. Цикл с предусловием. Цикл с постусловием.	10	2
	<i>Лабораторные работы.</i> Работа с текстовыми файлами.	4	3
Тема 2.3 Массивы,	<i>Содержание учебного материала</i>		

строки и указатели	<i>Лекционные занятия.</i> Одномерные и многомерные массивы. Символы и строки.	4	1
	<i>Практические занятия.</i> Одномерные массивы. Матрицы.	4	2
Тема 2.4 Функции	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Объявление и вызов функций. Передача аргументов в функции. Перегрузка и шаблоны функции.	4	1
	<i>Лабораторные работы.</i> Перегрузка функций. Шаблоны функций. Рекурсии в C++.	8	3
Всего:		59	

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально – техническому обеспечению

Лаборатория разработки информационных систем

12 персональных компьютеров; проектор View Sonic PG603X DLP; экран настенный Lumien

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

Mozilla Firefox (MPL)

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

QT Creator ((L)GPL)

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, интернет – ресурсов, дополнительной литературы.

Основные источники:

1. Павловская Т.А. Программирование на языке высокого уровня С# [Электронный ресурс]/ Павловская Т.А.— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 245 с..
<http://www.iprbookshop.ru/73713.html>
2. Баженова И.Ю. Введение в программирование [Электронный ресурс]: учебное пособие/ Баженова И.Ю., Сухомлин В.А.— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.— 326 с..
<http://www.iprbookshop.ru/97539.html>
3. Кирнос, В. Н. Информатика 2. Основы алгоритмизации и программирования на языке С++ : учебно-методическое пособие / В. Н. Кирнос. — Томск : Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2013. — 160 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. .
<https://www.iprbookshop.ru/14011.html>

Дополнительные источники:

1. Канунова Е.Е., Стулов Н. Н Информатика и программирование: учебное пособие. В 2-х частях [Гриф]. Ч.1. / Канунова Е.Е., Стулов Н. Н, Стулов Н. Н - Муром: ИПЦ МИ ВлГУ, 2008. - 74с.. 60
2. Канунова Е.Е., Стулов Н. Н Информатика и программирование: учебное пособие. В 2-х частях [Гриф]. Ч.2. / Канунова Е.Е., Стулов Н. Н, Стулов Н. Н - Муром: ИПЦ МИ ВлГУ, 2008. - 72с. . 60
3. Золин, А. Г. Программирование на С++ : учебное пособие для СПО / А. Г. Золин, А. Е. Колоденкова, Е. А. Халикова. — Саратов : Профобразование, 2022. — 126 с. — ISBN 978-5-4488-1439-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. <https://www.iprbookshop.ru/14011.html>

Интернет-ресурсы:

1. Электронная библиотечная система - iprbookshop.ru
2. Электронная библиотека ВлГУ - e.lib.vlsu.ru

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
т применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий	контрольная работа, устный опрос
Демонстрирует знания алгоритмизации, языков и технологий программирования, пригодных для практического применения в области информационных систем и технологий	контрольная работа, устный опрос

Фонд оценочных материалов (средств) по дисциплине
Основы алгоритмизации и программирования

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

1. Темы для устного опроса:

Рейтинг-контроль №1

- Основные этапы решения задач на ЭВМ
- Уровни и методы проектирования программ
- Структурное программирование
- Основные свойства алгоритма
- Способы записи алгоритма
- Вербальное представление алгоритма.
- Построчная запись алгоритма
- Представление алгоритма в виде блок-схемы
- Представление алгоритма на языке программирования
- Языки программирования
- Системы программирования
- История возникновения языка C++
- Алфавит и идентификаторы
- Правила образования идентификаторов
- Ключевые слова
- Символы операций и разделители
- Литералы
- Основные типы данных
- Логический тип (bool)
- Символьные типы (char, wchar_t)
- Целые типы (short, int, long)
- Вещественные типы (float, double, long double)
- Тип void
- Объявление переменных
- Время существования и область видимости переменных
- Комментарии
- Операции, выражения и операторы
- Арифметические операции
- Инкремент и декремент
- Операции отношения
- Логические операции
- Операции присваивания
- Побитовые операции
- Сдвиги
- Приведение типов
- Ключевое слово typedef
- Перечисления

Рейтинг-контроль №2

- Встроенные потоки ввода–вывода
- Форматированный ввод-вывод
- Форматирование с использованием флагов
- Использование манипуляторов ввода-вывода
- Установка ширины поля, точности и символов заполнения
- Функции ввода-вывода printf и scanf

- Ветвления и циклы
- Операторы выбора
- Условный оператор (if)
- Тернарный оператор ?:
- Оператор switch
- Операторы цикла
- Цикл for
- Цикл while
- Цикл do...while
- Операторы управления циклами – break и continue
- Оператор безусловного перехода goto
- Одномерные массивы
- Многомерные массивы
- Указатели
- Адресная арифметика
- Указатели и массивы
- Строки
- Ввод строк с клавиатуры
- Функции работы со строками
- Работа с динамической памятью
- Динамические массивы
- Ссылки
- Объявление, определение и вызов функции
- Передача аргументов
- Передача аргументов по значению
- Передача аргументов через указатель
- Передача аргументов по ссылке
- Аргументы по умолчанию
- Рекурсивные функции
- Встраиваемые функции
- Перегрузка функций
- Шаблоны функций

Рейтинг-контроль №3

- Директива #include
- Директива #define
- Директивы условной компиляции
- Основные принципы ООП
- Объявление класса
- Использование класса. Объекты
- Конструкторы
- Деструкторы
- Структуры
- Объединения
- Генерация и перехват исключений
- Генерация исключений внутри функций
- Повторная генерация исключения

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Контрольная работа, практические работы, лабораторные работы	20
Рейтинг-контроль 2	Контрольная работа, практические работы, лабораторные работы	20
Рейтинг-контроль 3	Контрольная работа, практические работы, лабораторные работы	20
Посещение занятий студентом		10
Дополнительные баллы (бонусы)		10
Выполнение семестрового плана самостоятельной работы		20

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

1. Темы для устного опроса:

Рейтинг-контроль №1

- Основные этапы решения задач на ЭВМ
- Уровни и методы проектирования программ
- Структурное программирование
- Основные свойства алгоритма
- Способы записи алгоритма
- Вербальное представление алгоритма.
- Построчная запись алгоритма
- Представление алгоритма в виде блок-схемы
- Представление алгоритма на языке программирования
- Языки программирования
- Системы программирования
- История возникновения языка C++
- Алфавит и идентификаторы
- Правила образования идентификаторов
- Ключевые слова
- Символы операций и разделители
- Литералы
- Основные типы данных
- Логический тип (bool)
- Символьные типы (char, wchar_t)
- Целые типы (short, int, long)
- Вещественные типы (float, double, long double)
- Тип void
- Объявление переменных
- Время существования и область видимости переменных
- Комментарии
- Операции, выражения и операторы

- Арифметические операции
- Инкремент и декремент
- Операции отношения
- Логические операции
- Операции присваивания
- Побитовые операции
- Сдвиги
- Приведение типов
- Ключевое слово typedef
- Перечисления

Рейтинг-контроль №2

- Встроенные потоки ввода–вывода
- Форматированный ввод-вывод
- Форматирование с использованием флагов
- Использование манипуляторов ввода-вывода
- Установка ширины поля, точности и символов заполнения
- Функции ввода-вывода printf и scanf
- Ветвления и циклы
- Операторы выбора
- Условный оператор (if)
- Тернарный оператор ?:
- Оператор switch
- Операторы цикла
- Цикл for
- Цикл while
- Цикл do...while
- Операторы управления циклами – break и continue
- Оператор безусловного перехода goto
- Одномерные массивы
- Многомерные массивы
- Указатели
- Адресная арифметика
- Указатели и массивы
- Строки
- Ввод строк с клавиатуры
- Функции работы со строками
- Работа с динамической памятью
- Динамические массивы
- Ссылки
- Объявление, определение и вызов функции
- Передача аргументов
- Передача аргументов по значению
- Передача аргументов через указатель
- Передача аргументов по ссылке
- Аргументы по умолчанию
- Рекурсивные функции
- Встраиваемые функции
- Перегрузка функций
- Шаблоны функций

Рейтинг-контроль №3

- Директива #include
- Директива #define
- Директивы условной компиляции
- Основные принципы ООП
- Объявление класса
- Использование класса. Объекты
- Конструкторы
- Деструкторы
- Структуры
- Объединения
- Генерация и перехват исключений
- Генерация исключений внутри функций
- Повторная генерация исключения

Методические материалы, характеризующих процедуры оценивания

При проведении промежуточных аттестаций используются вопросы, приведенные в пункте "Оценочные средства для промежуточной аттестации". Из каждого раздела, освоенного студентом, выбирается по два теоретических и одному практическому вопросу. Теоретические вопросы раскрываются в устной, либо в письменной форме. Практические задания как правило реализуются с помощью персонального компьютера.

При проверке знаний, приобретенных в рамках выполнения практических и лабораторных работ, используются контрольные вопросы, приведенные в методических указаниях к практическим работам. Защита практических и лабораторных работы также является средством промежуточной аттестации.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	<i>Уровень сформированности компетенций</i>
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	<i>Высокий уровень</i>
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	<i>Продвинутый уровень</i>

50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<i>Пороговый уровень</i>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<i>Компетенции не сформированы</i>

3. Задания в тестовой форме по дисциплине

Примеры заданий:

- Расположите типы данных в порядке невозрастания их размеров
 - char
 - float
 - double
 - long double
- Расположите типы данных в порядке убывания их размеров
 - char
 - float
 - double
 - long double
- Укажите назначение функций обработки строк
 - strcpy
 - strcmp
 - strlen
 - strchr
- Укажите размеры типов данных в компиляторе Visual Studio 32bit
 - char
 - short
 - int
 - double
- Укажите тип приведённых литералов
 - 0xFF
 - 2e3
 - false
 - 'a'
 - "true"
- Укажите тип приведённых литералов
 - 0x85
 - 3.0
 - true
 - '1'
 - "1"
- В каких вариантах допущены ошибки
 - char str[20]; str = "Hello";

- `char str[20] = "Hello";`
 - `char str[50]; strcpy("Hello", str);`
 - `char str[5]; strcpy(str, "Hello, world");`
 - `char str[2]; strcpy(str, "3");`
8. В каких случаях будет истинным условие `if (!strcmp(str1, str2))`
- Строки `str1` и `str2` равны
 - Строки `str1` и `str2` не равны
 - Строка `str1` "больше" строки `str2`
 - Строка `str1` "меньше" строки `str2`
 - Такая запись недопустима
9. В каких случаях доступ к элементам массива осуществляется неправильно?
- `int a[10]; a[10];`
 - `short a[5]; a[24%5];`
 - `int a[3][4]; a[2];`
 - `int a[5]; a[2][2];`
10. В каких случаях инициализация массива осуществляется без синтаксических ошибок?
- `int a[2] = {1, 2, 3};`
 - `float a[3] = {2.0, 3, 4};`
 - `short b[4] = {2};`
 - `char a[4] = "12";`
 - `double d[3]; d = {1.2, 2.4, 3.4};`

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/course/view.php?id=3248>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.