

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Колледж радиоэлектроники имени П.Н. Яблочкова



Рабочая программа учебной дисциплины

Основы алгоритмизации и программирования

09.02.06 Сетевое и системное администрирование

Профиль подготовки
технологический
Квалификация выпускника
сетевой и системный администратор
Форма обучения
очная

Саратов

2020


Разработчики: преподаватель З.И. Дервянченко



Рассмотрено на заседании ЦК программирования, информатики и
вычислительной техники

от «25» 05. 2020 г. протокол № 9

Председатель ЦК программирования, информатики и вычислительной
техники



Е.Д. Шаманаева

Директор Колледжа
радиоэлектроники
имени П.Н.Яблочкова



О.В. Бреус

Заместитель директора по УР



Н.Н. Чернова

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.06 Сетевое и системное администрирование.

Организация - разработчик: ФГБОУ ВО «СГУ имени Н.Г. Чернышевского» Колледж радиоэлектроники имени П.Н. Яблочкова СГУ.

Разработчик: Деревянченко З.И. - преподаватель Колледжа радиоэлектроники имени П.Н. Яблочкова СГУ.

СОДЕРЖАНИЕ

	стр.
1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	14
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	15

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

1.1. Область применения программы

Программа учебной дисциплины является частью основной профессиональной образовательной программы в соответствии с ФГОС СПО по специальности 09.02.06 Сетевое и системное администрирование.

1.2. Место дисциплины в структуре программы подготовки специалистов среднего звена: дисциплина относится к общеобразовательному учебному циклу.

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

В результате освоения дисциплины обучающийся должен уметь:

- разрабатывать алгоритмы для конкретных задач;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовать построенные алгоритмы в виде программ на конкретном языке программирования;
- оформлять код программы в соответствии со стандартом кодирования;
- выполнять проверку и отладку кода программы.

В результате освоения дисциплины обучающийся должен знать:

- 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-ориентированного программирования на примере алгоритмического языка.

ПК и ОК, которые актуализируются при изучении учебной дисциплины:

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 3. Планировать и реализовывать собственное профессиональное и личностное развитие.

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.

ОК 7. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.

ОК 8. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.

ОК 9. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках.

ОК 11. Планировать предпринимательскую деятельность в профессиональной сфере.

ПК 1.2. Осуществлять выбор технологии, инструментальных средств и средств вычислительной техники при организации процесса разработки и исследования объектов профессиональной деятельности.

ПК 2.3. Обеспечивать сбор данных для анализа использования и функционирования программно-технических средств компьютерных сетей.

ПК 2.4. Взаимодействовать со специалистами смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности.

1.4. Количество часов на освоение программы дисциплины:

учебной нагрузки обучающегося 138 часов, в том числе:

учебной нагрузки во взаимодействии с преподавателем 114 часов;

самостоятельной работы обучающегося 16 часов

промежуточная аттестация 8 часов

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Учебная нагрузка (всего)	138
Учебная нагрузка во взаимодействии с преподавателем	114
в том числе:	
лабораторные занятия	60
практические занятия	-
контрольные работы	-
курсовая работа (проект)	-
консультации и экзамены	8
Самостоятельной работы обучающегося (всего)	16
в том числе:	
Самостоятельная работа над курсовой работой (проект)	-
Промежуточная аттестация в форме экзамена	

2.2 Тематический план и содержание учебной дисциплины Основы алгоритмизации и программирования

Наименование занятий	Содержание учебного материала, лабораторные и практические занятия, внеаудиторная (самостоятельная) учебная работа (проект).	Объем часов	Уровень усвоения
1	2	3	4
Введение. Роль дисциплины в становлении специалистов.	1. Роль дисциплины в становлении специалистов. 2. Новые концепции программирования. 3. Классы программистов.	2	1
Раздел 1 Основы теории алгоритмов		4	
	Содержание	4	
Тема 1.1. Понятие и свойства алгоритма. Типы алгоритмов. Обозначение блоков алгоритма.	1. Понятие алгоритма. 2. Пример простого алгоритма. 3. Свойства алгоритма. 4. Представление алгоритма в виде блок-схемы.	2	1
	Самостоятельные занятия студентов	2	
	Тематика самостоятельной работы (конспект) Работа с блок-схемами.		3
Раздел 2. Основные понятия языка С++		48	
	Содержание	2	
Тема 2.1. История развития С и С++.	1. Основные понятия языка С и С++. 2. Языки низкого и высокого уровня. 3. Концепция памяти. 4. Этапы решения задачи в С++	2	1
	Содержание	6	
Тема 2.2. Состав языка	1. Алфавит языка. 2. Идентификаторы. 3. Ключевые слова 1. Классификация типов данных. 2. Описание простых типов данных. 3. Концепция памяти. Переменные 4. Пример простой программы.	2	1
		2	1

	Самостоятельные занятия студентов Тематика самостоятельной работы Глоссарий на тему «Изучение интегрированной среды разработки VisualStudio 10 Express»	2	3
	Содержание	2	
Тема 2.3. Структура программы языка C++	1. Структура программы. 2. Директивы процессов. 3. Объявления и определения. 4. Макроподстановки.	2	1
	Содержание	4	
Тема 2.4. Организация ввода и вывода информации	1. Стандартный и потоковый ввод – вывод. 2. Операторы ввода-вывода. Формат записи операторов. 3. Формат вывода целых и действительных чисел.	2	1
	Лабораторная работа №1 Программирование алгоритмов линейной структуры.	2	2
	Содержание	4	
Тема 2.5. Арифметические операции языка C++	1. Классификация операций. 2. Одноместные и двухместные операции. 3. Использование в выражениях операторов разных типов. 4. Дополнительные операции. 5. Приоритет выполняемых действий в выражении.	2	1
	Лабораторная работа №2 Программирование алгоритмов линейной структуры со стандартными функциями	2	2
	Содержание	2	
Тема 2.6. Функции языка C++	1. Основные понятия. 2. Локальные и глобальные переменные. 3. Параметры функций. 4. Классы памяти. 5. Примеры использования функций.	2	1
	Содержание	8	1

Тема 2.7. Условный оператор, оператор выбора, оператор перехода. Программирование задач с разветвлением.	1. Простые операторы языка C++.	2	
	2. Оператор изменяющие естественный ход выполнения программы.		
	3. Условный оператор, оператор выбора и безусловный оператор.		
	4. Формат записи и особенности использования		
	Лабораторная работа №3 Программирование алгоритмов с разветвляющейся структурой.	4	2
		Лабораторная работа №4 Программирование алгоритмов с разветвляющейся структурой.	
		Самостоятельная работа	
	Тематика внеаудиторной самостоятельной работы (индивидуальные задания). Освоение основных приемов работы с оператором выбора.	2	3
		Содержание	
		1. Организация циклических процессов.	
2. Операторы цикла языка C++.			
Тема 2.8. Операторы Цикла. Программирование циклических задач.	3. Решение задач итерационной структуры.	4	1
	1. Решение задач по вычислению определенного интеграла.		
	2. Вычисление длины кривой.		
	3. Локализация корней алгебраических уравнений		
Лабораторные работы	12	2	
Лабораторная работа №5 Программирование алгоритмов циклической структуры с заданным числом повторений			
Лабораторная работа №6 Программирование алгоритмов циклической структуры с внутренней разветвляющейся структурой.			
Лабораторная работа №7 Программирование алгоритмов итерационной циклической структуры.			
Лабораторная работа №8 Программирование алгоритмов вычисления определенных интегралов (вычисление длины кривой).			

	Самостоятельная работа	4	3
Раздел 3. Сложные типы данных	Тематика внеаудиторной самостоятельной работы (индивидуальные задания) Программирование циклических задач.	62	
Тема 3.1 Массивы.	<p>Содержание</p> <ol style="list-style-type: none"> 1. Указатели. Ссылки. Одномерные массивы. 2. Выполнение операций над элементами массива. 3. Основные типы задач, с использованием одномерных массивов. <ol style="list-style-type: none"> 1. Сортировка массивов. 2. Классические методы сортировки. <p>Улучшенные методы сортировки.</p> <p>Многомерные массивы. Основные типы задач, с использованием многомерных массивов.</p> <p>Лабораторные работы</p> <p>Лабораторная работа №10 Программирование алгоритмов обработки одномерного массива.</p> <p>Лабораторная работа №11 Сортировка массивов</p> <p>Лабораторная работа №12 Программирование алгоритмов обработки многомерных массивов.</p> <p>Лабораторная работа №13 Программирование алгоритмов обработки многомерных массивов.</p> <p>Самостоятельные занятия студентов</p> <p>Тематика самостоятельной работы</p> <p>Сравнительный анализ различных реализаций быстрой сортировки массива.</p> <p>Тематика внеаудиторной самостоятельной работы (индивидуальные задания) Программирование алгоритмов обработки массивов.</p> <p>Содержание</p> <p>Основные операции с матрицами.</p>	22	1
		10	2
Тема 3.2. Обработка матриц.		6	1

	Лабораторная работа №14 Программирование обработки матриц.	4	2
	Содержание	6	
Тема 3.3. Битовые операции.	Побитовые операции с матрицами.	2	1
	Лабораторная работа №15 Работа с битовыми операциями.	4	
	Содержание	6	
	Программирование обработки строк на языке C++	2	1
	Лабораторные работы	4	
Тема 3.4. Строковый тип данных.	Лабораторная работа №16 Программирование алгоритмов обработки последовательности символов		2
	Лабораторная работа №16 Программирование алгоритмов обработки последовательности символов		
	Содержание	10	
	1. Файлы. 2. Классификация файлов. 3. Последовательность работы с файлами. 4. Стандартные функции и процедуры для работы с файлами и каталогами.	8	1
	1. Последовательность работы с типизированными файлами. 2. Последовательность работы с текстовыми файлами.		
Тема 3.5. Файлы. Последовательность работы с файлами.	Лабораторная работа №17 Программирование и работа с файлами.		2
	Самостоятельные занятия студентов	2	
	Тематика самостоятельной работы Изучение критических и фатальных ошибок программы в Visual Studio 10 Express.		3
	Содержание	6	
Тема 3.6 Работа со структурами.	1. Структура. Описание и работа. 2. Выполнение операций над полями структуры.	2	1

	Лабораторная работа №18 Программирование и работа со структурами.	4	2
	Содержание	6	
Тема 3.7. Указатели и динамические массивы.	1. Указатели. Работа с памятью.	2	1
	2. Динамические массивы. Освобождение памяти.		
	Лабораторная работа №19 Программирование с использованием указателей и динамических массивов.	4	2
Раздел 4. Оформление программы на языке C++		14	
	Содержание	14	
Тема 4.1 Оформление подпрограмм	1. Оформление подпрограмм.	8	1
	2. Процедуры и функции. Описание и работа		
	3. Рекурсии		
	Рекуррентные обращения к подпрограмме.		1
	Шаблон функции		1
	Лабораторные работы	6	
	Лабораторная работа №20 Программирование алгоритмов с использованием функций.		2
	Лабораторная работа №21 Программирование алгоритмов с использованием рекурсии.		2
	Лабораторная работа №22 Программирование с использованием шаблона функций.		2
	Промежуточная аттестация в виде экзамена	8	
Всего	138		

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. – ознакомительный (узнавание ранее изученных объектов, свойств);
2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

3. Условия реализации рабочей программы учебной дисциплины

3.1. Техническое обеспечение

Реализация программы дисциплины требует наличия лаборатории вычислительной техники, архитектуры персонального компьютера и периферийных устройств;

Технические средства обучения: интерактивная доска с мультимедийным проектором, персональный компьютер для преподавателя, рабочие станции для проверки знаний студентов.

Оборудование лаборатории и рабочих мест лаборатории: компьютерные рабочие станции для работы студентов.

3.2. Информационное обеспечение обучения

Основные источники

1. Лафоре Роберт. Объектно-ориентированное программирование в C++: пер. с англ. /Р. Лафоре. - 4-е изд. - Москва ; Санкт-Петербург [и др.] : Питер, 2017. - 923 с. : ил.
2. Огнева М.В. Структуры данных и алгоритмы: программирование на языке C++: учебное пособие: в 2 ч. / М. В. Огнева, Е. В. Кудрина. - Саратов: Издательский центр "Наука", 2013 - .Ч. 1. - Саратов : Издательский центр "Наука", 2016. - 87 с.
3. В.Д. Колдаев; Под ред. Л.Г. Гагариной. – М. Основы алгоритмизации и программирования: Учебное пособие: ИД ФОРУМ: ИНФА-М. 2016 – 416с.

Дополнительные источники

4. Голицына, О. Л., Попов И.И. Основы алгоритмизации и программирования: Учебное пособие. – М.: Форум: ИНФРА-М, 2015. – 432 с.
5. Семакин И.Г. Основы программирования: Учебник для сред. проф. образования/И.Г.Семакин, А.П.Шестаков. – 3-е изд., стер. – М.: Издательский центр «Академия», 2015. – 432с.
6. Павловская Т.А. С/С++. Программирование на языке высокого уровня. СПб.: Питер, 2017. – 461 с.
7. Хабибулин И.Ш. Программирование на языке высокого уровня. С/С++. – СПб.: БХВ–Петербург, 2016. – 512 с.
8. Тихомиров Ю. VisualC++ – СПб.: БХВ–Санкт-Петербург, 1999. – 496 с.

Электронные ресурсы

1. Кауфман, В. Ш. Языки программирования. Концепции и принципы [Электронный ресурс] : учебное пособие / Кауфман В. Ш. - Москва : ДМК Пресс, 2011. - 464 с.

2. Персональная страничка Диканева Тараса Викторовича
Программирование для начинающих, 2011 [Электронный ресурс]. –
Режим доступа: <http://www.tvd-home.ru/prog>

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

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

Результаты обучения (освоенные умения, усвоенные знания)	Основные показатели оценки результата
<p>общие принципы построения алгоритмов, основные алгоритмические конструкции;</p> <p>эволюцию языков программирования, их классификацию, понятие системы программирования;</p> <p>основные элементы языка, структуру программы, операторы и операции, управляющие структуры;</p> <p>структуры данных, файлы, классы памяти;</p> <p>подпрограммы, составление библиотек подпрограмм;</p> <p>объектно-ориентированного программирования на примере алгоритмического языка.</p> <p>разрабатывать алгоритмы для конкретных задач;</p> <p>определять сложность работы алгоритмов;</p> <p>работать в среде программирования;</p> <p>реализовать построенные алгоритмы в виде программ на конкретном языке программирования;</p> <p>оформлять код программы в соответствии со стандартом кодирования;</p> <p>выполнять проверку и отладку кода программы.</p>	<p>понимание общих принципов построения алгоритмов, основных алгоритмических конструкций;</p> <p>понимание эволюции языков программирования, их классификации, понятия системы программирования;</p> <p>понимание основных элементов языка, структуры программы, операторов и операций, управляющих структур;</p> <p>понимание структуры данных, файлов, классов памяти;</p> <p>понимание написания подпрограммы, составление библиотек подпрограмм;</p> <p>понимание объектно-ориентированного программирования на примере алгоритмического языка.</p> <p>анализ разработки алгоритмов для конкретных задач;</p> <p>умение определять сложность работы алгоритмов;</p> <p>работать в среде программирования;</p> <p>реализовать построенные алгоритмы в виде программ на конкретном языке программирования;</p> <p>умение оформлять код программы в соответствии со стандартом кодирования;</p> <p>выполнять проверку и отладку кода программы.</p>