

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Худин Александр Николаевич

Должность: Ректор

Дата подписания: 29.01.2021 11:05:59

Уникальный программный ключ:

08303ad8de1c60b987361de7085acb509ac3da143f415362ffaf0ee37e73fa19

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Курский государственный университет»

Колледж коммерции, технологий и сервиса

УТВЕРЖДЕНО
протокол заседания
ученого совета от 07.04.2020 г., №8

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования



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).

Организация – разработчик: ФГБОУ ВО «Курский государственный университет».

Разработчик:

Бобрышева В.В. – преподаватель колледжа коммерции, технологий и сервиса ФГБОУ ВПО «Курский государственный университет».

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	стр. 3
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	7
3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	11
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	13

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

1.1. Область применения программы

Рабочая программа учебной дисциплины является частью ППССЗ в соответствии с ФГОС по специальности СПО **09.02.05 Прикладная информатика (по отраслям)**.

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании (в программах повышения квалификации и переподготовки).

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

дисциплина входит профессиональный цикл

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

Процесс изучения учебной дисциплины направлен на формирование следующих компетенций:

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 6	Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями
ОК 7	Брать на себя ответственность за работу членов команды (подчиненных), результат выполнения заданий
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 1.1	Обрабатывать статический информационный контент
ПК 2.1	Осуществлять сбор и анализ информации для определения потребностей клиента

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 отраслевой направленности
ПК 2.4	Проводить адаптацию отраслевого программного обеспечения
ПК 3.1	Разрешать проблемы совместимости программного обеспечения отраслевой направленности
ПК 4.1	Обеспечивать содержание проектных операций
ПК 4.4	Определять ресурсы проектных операций

В результате освоения дисциплины обучающийся должен **уметь**:

- составлять программы с ветвлениями;
- составлять циклические программы;
- осуществлять ввод и вывод массивов;
- сравнивать массивы;
- выполнять операции со строками;
- применять стандартные функции для работы с массивами и строками;
- объявлять структуры и объединения и использовать их в программе;
- использовать функции в программах;
- создавать программы с использованием функций;
- создавать собственные функции;
- работать с файлами разных типов;
- программировать модули;
- использовать модули;
- описать заголовки и реализацию класса;
- использовать объекты класса;
- создавать класс-наследник данного класса;
- реализовывать разные виды конструкторов и деструкторов;
- создавать и использовать виртуальные члены-функции класса;
- создавать чисто абстрактные классы;
- переопределять функции и операторы;
- реализовывать и использовать статические члены-данные и члены-функции;

В результате освоения дисциплины обучающийся должен **знать**:

- иметь представление о роли и месте знаний по дисциплине в процессе освоения основной профессиональной образовательной программы по специальности;
- понятие алгоритма;
- свойства алгоритма;
- базовые структуры алгоритмов;

- основные базовые типы данных;
- основные структурированные типы данных;
- основные операции логической алгебры;
- основные свойства логических операций;
- понятия рекурсии, рекурсивных функций и алгоритмов;
- алгоритмы сортировки;
- примеры рекурсивных вычислительных алгоритмов;
- классификацию языков программирования;
- понятие системы программирования;
- алфавит и лексику языка C++;
- типы данных языка программирования C++;
- правила записи выражений и операций;
- функции ввода/вывода;
- интерфейс среды программирования;
- порядок работы с командами меню и инструментами;
- директивы препроцессора C;
- оператор присваивания;
- оператор ветвления;
- оператор вывода;
- операторы цикла;
- синтаксис объявления массива, строки;
- выполняемые операции с массивами, строками;
- функции для работы со строками модуля <cstring>;
- понятие ссылки;
- понятие указателя;
- понятие подпрограммы;
- синтаксис объявления функций;
- стандартные функции библиотек языка C++;
- основные определения методов доступа;
- синтаксис объявления файла;
- порядок и правила работы с файлами;
- назначение модулей;
- правила подключения заголовочных файлов;
- принципы сокрытия данных и алгоритмов в файлах реализации;
- синтаксис описания класса;
- уровни инкапсуляции класса;
- основные отличия класса от структуры и объединения;
- порядок создания экземпляров класса;
- основные принципы наследования классов;
- реализацию наследования классов;
- виды конструкторов, их назначение и синтаксис их описания;
- назначение деструкторов и синтаксис их описания;

- способы переопределения функций;
- способы переопределения операторов.

1.4. Рекомендуемое количество часов на освоение программы дисциплины:

максимальной учебной нагрузки обучающегося 146 часов, в том числе:

обязательной аудиторной учебной нагрузки обучающегося 16 часов;
самостоятельной работы обучающегося 130 часов.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	146
Обязательная аудиторная учебная нагрузка (всего)	16
в том числе:	
лабораторные занятия	-
практические занятия	8
контрольные работы	-
Самостоятельная работа обучающегося (всего)	130
Оформление плана-конспекта	14
Подготовка реферата	32
Индивидуальные проектные задания	32
Оформление отчета по практическим работам	32
Оформление контрольной работы	20
Итоговая аттестация в форме <i>экзамена</i>	

2.2. Тематический план и содержание учебной дисциплины «Основы алгоритмизации и программирования»

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, курсовая работа (проект)	Объем часов	Уровень освоения
1	2	3	4
Раздел 1. Основные принципы алгоритмизации и программирования		34	
Тема 1.1. Основные принципы алгоритмизации и программирования	Содержание	34	
	1 Введение в дисциплину. Алгоритмизация. Основные понятия. Базовые алгоритмические конструкции. Типы данных. Структуры данных. Абстрактные типы данных	2	2
	Самостоятельная работа обучающихся Примерная тематика внеаудиторной работы: Понятие алгоритма. Свойства алгоритмов. Формы записи алгоритмов. Общие принципы построения алгоритмов. Основные алгоритмические конструкции: линейные, разветвляющиеся, циклические Данные: понятие и типы. Основные базовые и структурированные типы данных, их характеристика Линейные структуры данных: списки, очереди, стеки. Древовидные структуры данных: виды деревьев, примеры использования древовидных структур Основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Минимизация высказываний с помощью законов логики. Понятие трансляции. Уровни трансляции. Языки программирования Классификация языков программирования. История развития языков программирования Этапы решения задач на ЭВМ	32	
Раздел 2. Программирование на алгоритмическом языке		40	

Тема 2.1. Программирование на алгоритмическом языке	Содержание		40	
	1	Алфавит языка. Идентификаторы. Структура программы. Базовые типы данных. Переменные и константы. Оператор присваивания. Преобразование типов. Логические операции. Операторы перехода и выбора Циклические операторы Алфавит языка C++. Идентификаторы. Служебные (ключевые) слова. Типы данных. Описание переменных. Константы. Запись вещественных констант. Использование суффиксов. Запись символьных и строковых констант. Именованные константы (константные переменные)	2	2
	Практические занятия		6	
	1	Составление программ линейной структуры		
	2	Составление программ разветвляющейся структуры		
	3	Составление программ циклической структуры		
	Самостоятельная работа обучающихся Примерная тематика внеаудиторной работы: Препроцессор языка C Code::Blocks: установка, настройка интерфейса Рекурсия и рекурсивные операции Понятие функции. Параметры функции. Работа с указателями Массивы. Алгоритмы работы с массивами Строки. Структуры и объединения Понятие файла. Физический файл. Дескриптор файла. Двоичные и текстовые файлы. Работа с файлами в C++ Модульное программирование Оформление отчета по практическим работам		32	
Раздел 3. Основы объектно-ориентированного программирования			70	
Тема 3.1. Понятие класса. Инкапсуляция данных и алгоритмов их обработки	Содержание		36	
	1	Объектно-ориентированное программирование. Предпосылки и история развития. Понятие класса. Инкапсуляция данных и алгоритмов их обработки. Описание класса. Поля класса. Особенности локального класса	2	2
	Практические занятия		2	
	1	Создание класса и использование его объектов.		

	Самостоятельная работа обучающихся Оформление отчета по практическим работам	32	
Тема 3.2. Наследование. Конструкторы, деструкторы. Виртуальные функции.	Содержание	36	
	1 Наследование и виртуальные функции Статические члены класса. Указатель this. Наследование. Конструкторы и деструкторы производных классов. Виртуальные функции. Абстрактные классы.	2	2
	Самостоятельная работа обучающихся Примерная тематика внеаудиторной работы: Группа. Способы реализации группы. Методы группы. Иерархия объектов. Итератор. Динамическая идентификация типов. Выполнение проектного задания. Абстрактный тип данных. Перегрузка операций. Перегрузка унарных операций. Перегрузка бинарных операций. Перегрузка операции присваивания. Оформление отчета по практическим работам	34	
	Всего:	146	

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация программы дисциплины требует наличия лаборатории разработки, внедрения и адаптации программного обеспечения отраслевой направленности.

Оборудование лаборатории:

- стол преподавателя – 1 шт.
- стол аудиторный двухместный – 17 шт.
- стулья аудиторные – 28 шт.
- компьютерные столы – 9 шт.
- доска аудиторная для написания мелом – 1 шт.
- табуретки – 6 шт.
- стеллаж – 1 шт.
- тумба – 1шт.
- трибуна – 1 шт.
- зеркало – 1 шт.

- комплект учебно-наглядных пособий по дисциплине «Основы алгоритмизации и программирования»;

Технические средства обучения:

- персональный компьютер в сборе - 13 шт.
- мобильный ПК (ноутбук) Asus M51V - 1 шт.
- проектор мультимедийный NEC np115 - 1шт.
- интерактивная доска Hitachi Star Board - 1 шт.
- МФУ лазерное Canon i-sensys MF 4410 - 1 шт.
- МФУ лазерное HP LaserJet M1132 MFP - 1 шт.

Программное обеспечение:

- Microsoft Windows Win10Pro (64) Акт приема-передачи от 31 июля 2017, контракт №0344100007517000020-0008905-01;

- Microsoft Windows XP Professional Open License: 47818817;
- Microsoft Office Professional Plus 2007 Open License:43219389;
- 7-Zip Свободная лицензия GNU LGPL;
- Adobe Acrobat Reader DC Бесплатное программное обеспечение;
- Google Chrome Свободная лицензия BSD;
- Code::Blocks Свободная лицензия GNU GPLv3;

1С: Предприятие 8. Комплект для обучения в высших и средних учебных заведениях. Договор № 040418 от 04.04.2018 г.;

- учебный комплект КОМПАС-3D V12 MCAD Лицензионное соглашение Кк-11-00122;

- Photoshop Extended Cs5 12.0 Win AOO Software License Certificate: 65049824;

- Audacity Свободная лицензия GNU GPL 2;
- VirtualDub Свободная лицензия GPL;
- MySQL Workbench Свободная лицензия GNU GPL;
- Far manager Свободная лицензия BSDL;
- Mozilla Firefox Свободное программное обеспечение - GNU GPL и GNU

LGPL.

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Трофимов, В. В. Основы алгоритмизации и программирования : учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2020. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <http://biblio-online.ru/bcode/454452>
2. Кудрина, Е. В. Основы алгоритмизации и программирования на языке С# : учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2020. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <http://biblio-online.ru/bcode/456221>
3. Кудинов Ю.И. Основы алгоритмизации и программирования [Электронный ресурс]: учебное пособие для СПО/ Кудинов Ю.И., Келина А.Ю.— Электрон. текстовые данные.— Липецк, Саратов: Липецкий государственный технический университет, Профобразование, 2020.— 71 с.— Режим доступа: <http://www.iprbookshop.ru/92834.html>.— ЭБС «IPRbooks»
4. Чурина Т.Г. Основы алгоритмизации и программирования [Электронный ресурс]: учебное пособие для СПО/ Чурина Т.Г., Нестеренко Т.В.— Электрон. текстовые данные.— Саратов, Москва: Профобразование, Ай Пи Ар Медиа, 2020.— 214 с.— Режим доступа: <http://www.iprbookshop.ru/96017.html>.— ЭБС «IPRbooks»

Дополнительные источники:

1. Огнева, М. В. Программирование на языке с++: практический курс : учебное пособие для СПО / М. В. Огнева, Е. В. Кудрина. — М. : Издательство Юрайт, 2017. — 335 с. — (Серия : Профессиональное образование). — ISBN 978-5-534-05780-5. — Режим доступа: www.biblio-online.ru/book/B76AB4A4-7623-4842-9136-B6ADC57B90BC.
2. Лубашева Т.В. Основы алгоритмизации и программирования [Электронный ресурс] : учебное пособие / Т.В. Лубашева, Б.А. Железко. — Электрон. текстовые данные. — Минск: Республиканский институт профессионального образования (РИПО), 2016. — 379 с. — 978-985-503-625-9. — Режим доступа: <http://www.iprbookshop.ru/67689.html>
3. Журналы: Программирование

Интернет-ресурсы:

1. Электронный учебник по С++: <http://cgui.ru>
2. Каталог книг по программированию на С, С++: <http://c2p.ru>

4. 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, проектов, исследований.

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<p><i>Умения:</i></p> <ul style="list-style-type: none"> – составлять программы с ветвлениями; – составлять циклические программы; – осуществлять ввод и вывод массивов; – сравнивать массивы; – выполнять операции со строками; – применять стандартные функции для работы с массивами и строками; – объявлять структуры и объединения и использовать их в программе; – использовать функции в программах; – создавать программы с использованием функций; – создавать собственные функции; – работать с файлами разных типов; – программировать модули; – использовать модули; – описать заголовок и реализацию класса; – использовать объекты класса; – создавать класс-наследник данного класса; – реализовывать разные виды конструкторов и деструкторов; – создавать и использовать виртуальные члены-функции класса; – создавать чисто абстрактные классы; – переопределять функции и операторы; – реализовывать и использовать статические члены-данные и члены-функции; <p><i>Знания:</i></p> <ul style="list-style-type: none"> – иметь представление о роли и месте знаний по дисциплине в процессе освоения основной профессиональной образовательной программы по специальности; – понятие алгоритма; – свойства алгоритма; – базовые структуры алгоритмов; – основные базовые типы данных; – основные структурированные типы данных; – основные операции логической алгебры; 	<p><i>Оценка выполнения практических работ.</i></p> <p><i>Оценка защиты рефератов, докладов, презентаций.</i></p> <p><i>Оценка выполнения индивидуальных заданий</i></p>

<ul style="list-style-type: none"> – основные свойства логических операций; – понятия рекурсии, рекурсивных функций и алгоритмов; – алгоритмы сортировки; – примеры рекурсивных вычислительных алгоритмов; – классификацию языков программирования; – понятие системы программирования; – алфавит и лексику языка C++; – типы данных языка программирования C++; – правила записи выражений и операций; – функции ввода/вывода; – интерфейс среды программирования; – порядок работы с командами меню и инструментами; – директивы препроцессора C; – оператор присваивания; – оператор ветвления; – оператор вывода; – операторы цикла; – синтаксис объявления массива, строки; – выполняемые операции с массивами, строками; – функции для работы со строками модуля <cstring>; – понятие ссылки; – понятие указателя; – понятие подпрограммы; – синтаксис объявления функций; – стандартные функции библиотек языка C++; – основные определения методов доступа; – синтаксис объявления файла; – порядок и правила работы с файлами; – назначение модулей; – правила подключения заголовочных файлов; – принципы сокрытия данных и алгоритмов в файлах реализации; – синтаксис описания класса; – уровни инкапсуляции класса; – основные отличия класса от структуры и объединения; – порядок создания экземпляров класса; – основные принципы наследование классов; – реализацию наследования классов; – виды конструкторов, их назначение и синтаксис их описания; – назначение деструкторов и синтаксис их описания; – способы переопределения функций; – способы переопределения операторов. 	<p style="text-align: center;"><i>Оценка ответов на устный опрос</i></p> <p style="text-align: center;"><i>Оценка выполнения тестовых заданий</i></p> <p style="text-align: center;"><i>Экзамен</i></p>
--	---