Федеральное государственное образовательное бюджетное
учреждение высшего образования
Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

УТВЕРЖДАЮ

Заместитель директора по
учебной работе

_____ Н.Ю. Долгова
«_26_» _июня_ 2023г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

09.02.07 Информационные системы и программирование

очно-заочная форма

Москва 2023г.

Рабочая программа учебный дисциплины разработана на основе федерального государственного образовательного стандарта среднего профессионального образования (далее – ФГОС СПО) по специальности 09.02.07 Информационные системы и программирование
(код и наименование специальности)

Разработчик:
Набиев Шахин Казымагомедович, преподаватель первой квалификационной категории Колледжа информатики и программирования

Рабочая программа учебной дисциплины рассмотрена и рекомендована к утверждению на заседании предметной (цикловой) комиссии информационных систем и программирования

Протокол от «11» мая 2023 г.№9

Председатель предметной (цикловой) комиссии _____ Н.Г. Титов

1. Общая характеристика рабочей программы дисциплины

1.1. Место дисциплины в структуре основной образовательной программы

Дисциплина «Основы алгоритмизации и программирования» является обязательной частью общепрофессионального цикла основной профессиональной образовательной программы в соответствии с ФГОС по специальности Информационные системы и программирование.

Цель и планируемые результаты освоения дисциплины:

В рамках программы учебной дисциплины студентами осваиваются умения и знания

Код общих и профессиональных компетенция	Умения	Знания
ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5	- 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 языке программирования С++; - оформлять код программы в соответствии со стандартом кодирования; - выполнять проверку, отладку кода программы.	- 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; свойств и методов; инкапсуляция и полиморфизма; наследования и переопределения.

2. Структура и содержание дисциплины

2.1. Объем дисциплины и виды учебной работы

Вид учебной работы	Объем в часах
Объем образовательной программы дисциплины	
Объем работы студентов во взаимодействии с преподавателем	
в том числе:	
теоретическое обучение	
практические занятия	
самостоятельная работа	
Консультации	
Промежуточная аттестация в форме экзамена	

Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности студентов	Объем в часах	Коды компетенций формированию которых способствует элемент программы.
	семестр		
РАЗДЕЛ 1. Введение в программирование			
Тема 1.1 Языки программирования	Содержание учебного материала		ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Развитие языков программирования. Обзор языков программирования.		
	Самостоятельная работа студентов		
	Области применения языков программирования Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы.		
	Жизненный цикл программы. Программа. Программный продукт и его характеристики.		
	Основные этапы решения задач на компьютере.		
	В том числе практических занятий		
Тема 1.2. Типы данных	Содержание учебного материала		
	Типы данных.		
	Самостоятельная работа студентов		
	Простые типы данных.		
	Производные типы данных. Структурированные типы данных.		
	В том числе практических занятий		
РАЗДЕЛ 2. Основы языка программирования С++.			
Тема 2.1. Операторы языка программирования	Содержание учебного материала		ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2
	Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор.		
	Самостоятельная работа студентов		

	Условный оператор и условная операция. Оператор выбора.		ПК 1.3 ПК 1.4
	Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы.		ПК 1.5 ПК 2.4 ПК 2.5
	Массивы. Двумерные массивы. Сортировка в массивах.		
	Структурированный тип данных – множество. Операции над множествами.		
	Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа		
	В том числе практических занятий		
	1. Практическое занятие «Составление программ линейной структуры»	2	
	2. Практическое занятие «Составление программ разветвляющейся структуры»	2	
	3. Практическое занятие «Составление программ циклической структуры»	2	
	4. Практическое занятие «Обработка одномерных массивов»	2	
	5. Практическое занятие «Обработка двумерных массивов».		
	6. Практическое занятие «Работа со строками».	2	
	7. Практическое занятие «Работа со строками».		
	8. Практическое занятие «Составление программ на файлы последовательного доступа».		
	семестр		
	9. Практическое занятие «Составление программ на типизированные файлы»		
	10. Практическое занятие «Составление программ на не типизированные файлы»		
Раздел 3. Подпрограммы в С++. Понятие модуля.			
Тема 3.1. Процедуры и функции	Содержание учебного материала	18	
	1. Общие сведения о подпрограммах. Определение и вызов подпрограмм.		ОК. 01 ОК. 02

	Область видимости и время жизни переменной.		ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Самостоятельная работа студентов	8	
	Механизм передачи параметров. Организация функций Рекурсия. Программирование рекурсивных алгоритмов.		
	В том числе практических занятий		
	.Практическое занятие «Подпрограммы в С++»		
	2.Практическое занятие «Организация процедур».		
	3.Практическое занятие «Организация функций».		
	4.Практическое занятие «Применение рекурсивных функций»		
Тема 3.2. Структуризация в программировании	Содержание учебного материала		ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Самостоятельная работа студентов	8	
	Основы структурного программирования. Методы структурного программирования		
	В том числе практических занятий		
	.Практическое занятие «Решение многоструктурных задач»		
Тема 3.3. Модульное программирование	Содержание учебного материала	24	
	Самостоятельная работа студентов	16	
	Основы структурного программирования. Методы структурного программирования Стандартные модули.	16	
	В том числе практических занятий	8	
	.Практическое занятие «Программирование модуля»	4	
	.Практическое занятие «Создание библиотеки подпрограмм».	4	
Раздел 4 Динамические данные		**16**	
Тема 4.1 Указатели.	Содержание учебного материала	16	

	1. Указатели. Описание указателей. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных.	2	ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Самостоятельная работа студентов	8	
	Структуры данных на основе указателей.		
	Задача о стеке.		
	В том числе практических занятий	6	
	Практическое занятие «Использование указателей для организации связанных списков».	4	
	.Практическое занятие «Изучение интегрированной среды разработчика».	2	
Раздел 5 Объектно-ориентированное программирование			
Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)	Содержание учебного материала	22	
	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	2	ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Самостоятельная работа студентов		
	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		
	3. Классы объектов. Компоненты и их свойства.	4	
	Событийно-управляемая модель программирования. Компонентно-ориентированный подход.	4	
	В том числе практических занятий	8	
	.Практическое занятие «Создание проекта с использованием компонентов для работы с текстом».	2	
	.Практическое занятие «Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени».	2	
	.Практическое занятие «События компонентов (элементов управления), их сущность и назначение».	2	

	.Практическое занятие «События компонентов (элементов управления), их сущность и назначение».	2	
Тема 5.2 Интегрированная среда разработчика.	Содержание учебного материала	1	ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Требования к аппаратным и программным средствам интегрированной среды разработчика	2	
	Самостоятельная работа студентов		
	Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов.	2	
	Панель компонентов и их свойства. Окно кода проекта.	2	
	Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.	4	
	Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. Настройка среды и параметров проекта.		
Тема 5.3. Визуальное событийно-управляемое программирование	Содержание учебного материала		
	Самостоятельная работа студентов		ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение.		
	Дополнительные элементы управления. Свойства компонентов. Виды свойств. Синтаксис определения свойств. Назначения свойств и их влияние на результат. Управление объектом через свойства.		
	События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий	2	
	В том числе практических занятий	10	
	.Практическое занятие «Создание процедур на основе событий».		

	.Практическое занятие «Создание проекта с использованием кнопочных компонентов».		
	.Практическое занятие «Создание проекта с использованием кнопочных компонентов».		
	.Практическое занятие «Создание проекта с использованием компонентов стандартных диалогов и системы меню».		
	.Практическое занятие «Создание проекта с использованием компонентов стандартных диалогов и системы меню».		
Тема 5.4 Разработка оконного приложения	Содержание учебного материала	1	ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	1. Разработка функционального интерфейса приложения. Создание интерфейса приложения.		
	Самостоятельная работа студентов		
	Разработка функциональной схемы работы приложения.		
	Разработка игрового приложения.		
	В том числе практических занятий		
	.Практическое занятие «Разработка оконного приложения с несколькими формами»		
Тема 5.5 Этапы разработки приложений	Содержание учебного материала		ОК. 01 ОК. 02 ОК. 04 ОК. 05 ОК. 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5
	1.Разработка приложения.		
	Самостоятельная работа студентов		
	Проектирование объектно-ориентированного приложения.		
	Создание интерфейса пользователя.		
	Тестирование, отладка приложения.		
	В том числе практических занятий		
	.Практическое занятие «Создание процедур обработки событий. Компиляция и запуск приложения».		
Тема 5.6 Иерархия классов	Содержание учебного материала		
	1. Классы ООП: виды, назначение, свойства, методы, события.		

	Самостоятельная работа студентов		
	2. Перегрузка методов.		
	3. Тестирование и отладка приложения.		
Консультация			
Промежуточная аттестация в форме экзамен			
Всего:			

3. Условия реализации дисциплины

Для реализации программы учебной дисциплины предусмотрен следующие специальные помещения (в соответствии с ФГОС и ПООП): лаборатория «Программирования и баз данных», оснащенная оборудованием:
Персональные компьютеры
Маркерная доска
Проектор
Экран
техническими средствами обучения:
Лицензионное программное обеспечение общего и профессионального назначения.

3.2. Информационное обеспечение реализации программы

Основные печатные и электронные издания:

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, 2022. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. —

Гниденко, И. Г. Технология разработки программного обеспечения: учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва: Издательство Юрайт, 2022. — 235 с. — (Профессиональное образование). — ISBN 978-5-534-05047-9. — Текст: электронный // Образовательная платформа Юрайт [сайт]. —

Адиль Бикеев: C++ - https://stepik.org

Дополнительные источники

Конова, Е. А. Алгоритмы и программы. Язык C++ : учебное пособие для спо / Е. А. Конова, Г. А. Поллак. — 2-е изд., стер. — Санкт-Петербург: Лань, 2021. — 384 с. — ISBN 978-5-8114-8576-5. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/177837 (дата обращения: 08.06.2022). — Режим доступа: для авториз. пользователей.

4. Контроль и оценка результатов освоения дисциплины

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

Результаты обучения	Критерии оценки	Методы оценки
Перечень знаний, осваиваемых в рамках дисциплины: - 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, 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, инкапсуляция и полиморфизма, наследования и переопределения; - основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения; **Перечень умений, осваиваемых в рамках дисциплины:**	«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко. «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками. «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки. «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.	- экспертная оценка выполнения практических работ; - выполнение самостоятельной работы, - решение практических задач; -промежуточная аттестация в форме экзамена.

- 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 языке программирования C++; оформлять код программы в соответствии со стандартом кодирования; выполнять проверку, отладку кода программы.		