

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский экономический университет имени Г.В. Плеханова»
Ереванский филиал**

РАБОЧАЯ ПРОГРАММА

учебной дисциплины ОП.14 Основы алгоритмизации и программирования

09.02.05 Прикладная информатика (по отраслям)

квалификация: техник-программист

РПУД актуализирована – 2021 г.
 Протокол заседания ПЦК № 9 от «28» апреля 2021г.
РПУД актуализирована – 2020 г.
 Протокол заседания ПЦК № 10 от «25» мая 2020г.
РПУД актуализирована – 2019 г.
 Протокол заседания ПЦК № 10 от «23» мая 2019г.

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский экономический университет имени Г.В. Плеханова»
Ереванский филиал**

РАБОЧАЯ ПРОГРАММА

учебной дисциплины ОП.14 Основы алгоритмизации и программирования

код специальности 09.02.05 Прикладная информатика (по отраслям)

форма обучения очная

РПУД актуализирована – 2020 г.
Протокол заседания ЦК № 10 от «25» мая 2020г.

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский экономический университет имени Г.В. Плеханова»
Ереванский филиал**

РАБОЧАЯ ПРОГРАММА

учебной дисциплины ОП.14 Основы алгоритмизации и программирования

код специальности 09.02.05 Прикладная информатика (по отраслям)

форма обучения очная

2019

СОГЛАСОВАНА:
Предметной (цикловой) методической
комиссией по дисциплинам
естественно-научного блока

Разработана на основе Федерального
государственного образовательного стандарта по
специальности среднего профессионального
образования 09.02.05 Прикладная информатика
по отраслям)

Протокол № 10 от «23» мая 2019г.

Председатель предметной (цикловой)
методической комиссии



_____ / Г.Б. Вирабян /
подпись

УТВЕРЖДЕНА:



Директор филиала _____ / Т.Р.Варданян /
подпись

Составители (авторы): Нахатакян С.Х., к.т.н доцент, кафедра
“Информационные технологии и гуманитарные
науки”



Рецензент: Мелконян А.А., канд. физ-мат. наук доцент РАУ, кафедра системного
программирования



СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	5
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	17
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	18

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины **ОП.14 Основы алгоритмизации и программирования** является частью ППССЗ (программы подготовки специалистов среднего звена) в соответствии с ФГОС СПО по специальности **09.02.05 «Прикладная информатика (по отраслям)»**.

1.2. Место дисциплины в структуре ППССЗ:

Учебная дисциплина **ОП.14 Основы алгоритмизации и программирования** входит в общепрофессиональный цикл.

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

В результате освоения дисциплины обучающийся должен

уметь:

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

знать:

1. Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
2. Эволюцию языков программирования, их классификацию, понятие системы программирования.
3. Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
4. Подпрограммы, составление библиотек подпрограмм.
5. 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.

Сформировать общие компетенции:

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.

ОК5. 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.

Сформировать профессиональные компетенции:

ПК-1.1. Обрабатывать статический информационный контент.

ПК-1.2. Обрабатывать динамический 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 2.5. Разрабатывать и вести проектную и техническую документацию.

ПК 2.6. Участвовать в измерении и контроле качества продуктов.

1.4. Рекомендуемое количество часов на освоение программы учебной дисциплины:

максимальная учебная нагрузка обучающегося	238	часа
включая:		
обязательная аудиторная учебная нагрузка	154	часа
самостоятельная работа	68	часа
ВСЕГО	238	часа

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Количество часов
Максимальная учебная нагрузка (всего)	238
Обязательная аудиторная учебная нагрузка (всего)	154
в том числе:	
лабораторные работы	
практические занятия	76
контрольные работы	
Консультации (всего)	16
Самостоятельная работа обучающегося (всего)	68
в том числе:	
тематика внеаудиторной самостоятельной работы	68
Итоговая аттестация в форме	Зачет с оценкой

2.2. Тематический план и содержание учебной дисциплины ОП.14 Основы алгоритмизации и программирования

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся, курсовая работа (проект) (если предусмотрены учебным планом)	Объем часов	Уровень освоения
1	2	3	4
Семестр I: Раздел 1. Введение в программирование			
Тема 1.1 Основы алгоритмизации	Содержание учебного материала	6	1
	1. Современное состояние и перспективы развития вычислительной техники		
	2. Место и роль программирования и алгоритмизации в современных компьютерных технологиях		
	3. Понятие алгоритмизации и алгоритма. Свойства алгоритмов.		
	4. Общие принципы построения алгоритмов. Способы описания алгоритмов.		
	5. Правила выполнения схем алгоритмов. Основные конструкции алгоритмического языка: линейный алгоритм, ветвление, цикл		
	6. Этапы решения задач с помощью ЭВМ: постановка задачи, создание модели, алгоритм, кодирование алгоритма, анализ результатов. Правила постановки задачи.		
	7. Способы описания алгоритмов: псевдокоды, блок-схема, программа.		
	8. Блок-схема: основные элементы, правила составления. Примеры.		
9. Составление блок-схем линейных алгоритмов. Составление блок-схем разветвляющихся алгоритмов			

	10. Составление блок-схем циклических алгоритмов.		
	практическая работа	6	
	самостоятельная работа Построить алгоритмы различной конструкции по индивидуальным заданиям. Проанализировать примеры алгоритмов (определение НОД 2-х чисел, раскладка шаров по цвету, выбор большего числа). Составление конспекта на тему «История развития термина алгоритма» WWW.wikipedia.org	4	
	консультация По теме самостоятельной работы	2	
Тема 1.2 Языки программирования	Содержание учебного материала	6	
	1. Развитие языков программирования. Обзор языков программирования		1
	2. Области применения языков программирования. Стандарты языков программирования.		
	3. Среда проектирования. Компиляторы и интерпретаторы.		
	4. Жизненный цикл программы программа. Программный продукт и его характеристики. Основные этапы решения задач на компьютере.		
	практическая работа	6	
самостоятельная работа	4		

Тема 1.3 Типы данных	Подготовка сообщения на тему: «Эволюция языков программирования» www.wikipedia.org . Составление опорного конспекта по теме «Понятие системы программирования, основные функции системы программирования».		
	консультация По теме самостоятельной работы	2	
	Содержание учебного материала	6	
	Переменные и константы. Объявление объектов данных. Внутреннее представление данных в памяти компьютера. Типы данных. Простые типы данных.		
	Производные типы данных. Структурированные типы данных. Язык Python и интерпретатор языка Python. Типизация в языке Python. Основные типы данных. Арифметические операторы и их обозначения в языке Python.		
	практическая работа	6	
	самостоятельная работа Проработка материала на тему: «Типы данных, определяемые программистом. Перечисляемый и интервальный типы». Анализ конспектов лекций.	4	
Раздел 2. Основные конструкции языков программирования.			
Тема 2.1 Операторы языка программирования	Содержание учебного материала	6	
	1. Структура программы на языке программирования Python. Операции и переменные. Приоритет выполнения операций. Операнды: литералы, выражения, переменные. Динамическая типизация. Инструкция присваивания. Запись.		1,2
	практическая работа	6	

	<p>самостоятельная работа Выполнение конспекта на тему: «Форматы вывода данных». Изучить приоритет выполнения действий в выражениях, подготовить конспект. Составление и отладка программ с использованием арифметических выражений. Решить учебные задачи содержащие ввод и вывод информации.</p>	4	
	<p>консультация По теме самостоятельной работы</p>	2	
<p>Тема 2.2 Написание простых программ.</p>	<p>Содержание учебного материала</p>	6	
	<p>1. Классификация ошибок в программах и методы их устранения. Комментарии и документирование, пояснение к шаблону. Форматирование текстовых строк. Модификаторы формата.</p>		
	<p>практическая работа</p>	6	
	<p>самостоятельная работа Подготовить рефераты по истории создания одного из языков программирования высокого уровня.</p>	4	
	<p>консультация По теме самостоятельной работы</p>	2	
<p>Тема 2.3 Управляющие структуры.</p>	<p>Содержание учебного материала</p>	6	1,2
	<p>Линейные алгоритмы. Правила записи арифметических выражений. Инструменты обработки целых чисел. Тип данных float(). Инструменты обработки вещественных чисел. Разветвляющие алгоритмы. Инструкция ветвления. Циклические алгоритмы. Виды циклов. Принцип работы цикла с параметром. Сложные циклические алгоритмы. Вложенные циклы.</p>		

Тема 2.4 Этапы решения задачи на компьютере.

практическая работа	6	
самостоятельная работа Составить программы линейного типа по индивидуальным заданиям. Составить программы с разветвлением по индивидуальным заданиям. Составить программы со сложным условием по индивидуальным заданиям. Составить программу, содержащую выбор по индивидуальным заданиям. Решить учебные задачи содержащие ввод и вывод информации. Составить программы с различными видами циклов по заданию. Ознакомиться с конструкцией "Цикл в цикле", составить блок-схему.	8	
консультация По теме самостоятельной работы	2	
Содержание учебного материала	7	1,2
Последовательное конструирование алгоритма.		
Этапы решения задачи на компьютере: постановка, формализация, создание алгоритма, программирование, тестирование и отладка.		
Этапы решения задачи на компьютере: программирование, тестирование и отладка.		
Словари .Ключ в словаре. Вхождение в словаре.		
Определение и обработка исключений.		
Командный способ организации систем управления.		

Пакетный процесс организации систем управления.		
Диалоговый способ организации систем управления.		
Принцип обратной связи – адаптивное управление.		
Каскадная модель разработки программного обеспечения.		
Введение в язык программирования Python. История и особенности языка программирования Python.		
Первый запуск среды разработки.		
Функции. Применение функций при решении задач.		
Линейные последовательности данных - кортежи.		
Линейные последовательности данных - списки.		
Обработка списков. Линейный поиск.		
Обработка списков. Двоичный поиск и сортировка.		
Реализация матриц на языке Python. Обработка матриц.		

Промежуточная аттестация	Метод последовательной детализации		
	Рекурсивные методы		
	Методы перебора в задачах поиска		
	Методы сортировки данных и сложность алгоритмов		
	практическая работа	6	
		Зачет с оценкой	
Семестр II: Раздел 3. Последовательности и файлы.			
Тема 3.1 Структуры данных. Работа с файлами.	Содержание учебного материала	17	1,3
	Сохраняемость данных.		
	Файлы и адреса файлов.		
	Типы файлов. Режимы открытия файла.		
	Этапы обработки файла: открытие, ввод/вывод данных, закрытие.		
	практическая работа	17	

	<p>самостоятельная работа Составить схему классификации файлов. Разработать пример файла с реальными данными и задание по обработке файла в программе. Подготовить сообщение о командах работы с типизированными файлами.</p>	20	
	<p>консультация По теме самостоятельной работы</p>	2	
Раздел 4. Основы объектно-ориентированного программирования.			
<p>Тема 4.1 Концепция объектно-ориентированного программирования.</p>	<p>Содержание учебного материала</p>	18	1,3
	Объектно-ориентированный анализ, проектирование, программирование.		
	Объекты и классы.		
	Программирование объектной модели.		
	Поля и методы класса.		
	Свойства и методы классов.		
	Взаимодействие экземпляров различных классов.		
	Программирование объектов и классов.		
	Определение инкапсуляции. Применение инкапсуляции в программировании.		

Принципы объектно-ориентированного программирования: наследование.		
Принципы объектно-ориентированного программирования: полиморфизм.		
практическая работа	17	
самостоятельная работа Изучение правил разработки интерфейса пользователя: рекомендаций и методов. Анализ логики и работы объектно-ориентированных программ. Современные тенденции в программировании. Перспективы программирования.	20	
консультация По теме самостоятельной работы	4	
Примерная тематика курсовой работы (проекта) (если предусмотрены)		
Самостоятельная работа обучающихся над курсовой работой (проектом) (если предусмотрены)		
Всего:	238	

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 ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация учебной дисциплины требует наличия:

Оборудование:

- посадочные места по количеству обучающихся;
- рабочее место преподавателя;
- учебно-методический комплекс учебной дисциплины.

Технические средства обучения:

- персональные компьютеры с лицензионным программным обеспечением.

Количество рабочих мест:

- Посадочные места по количеству обучающихся;
- Рабочее место преподавателя

3.2. Информационное обеспечение обучения

Перечень учебных изданий, Интернет-ресурсов, дополнительной литературы

№ п/п	Наименование учебных изданий, Интернет-ресурсов, дополнительной литературы
I	Основные источники
1	<i>Трофимов, В. В.</i> Основы алгоритмизации и программирования : учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2018. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/422888
II	Дополнительные источники
1	<i>Кудрина, Е. В.</i> Основы алгоритмизации и программирования на языке С# : учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2019. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/431505
III	Интернет-ресурсы
1	https://academia-moscow.ru/ftp_share/books/fragments/fragment_20547.pdf
2	https://codernet.ru/books/python/izuchaem_python_4-e_izdanie_mark_lutc/

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Образовательное учреждение, реализующее подготовку по учебной дисциплине, обеспечивает организацию и проведение промежуточной аттестации и текущего контроля индивидуальных образовательных достижений - демонстрируемых обучающимися знаний, умений и навыков.

Текущий контроль проводится преподавателем Нахатакян С.Х.

Обучение по учебной дисциплине завершается промежуточной аттестацией в виде зачет с оценкой

Формы и методы промежуточной аттестации и текущего контроля по учебной дисциплине самостоятельно разрабатываются образовательным учреждением и доводятся до сведения обучающихся не позднее начала двух месяцев от начала обучения.

Итоговой формой контроля является зачет с оценкой

Фонды оценочных средств (ФОС, КОС) разрабатываются образовательным учреждением. Они включают в себя педагогические контрольно-оценочные материалы, предназначенные для определения соответствия (или несоответствия) индивидуальных образовательных достижений основным показателям результатов подготовки (таблицы).

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<p>умения</p> <ul style="list-style-type: none"> • 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. 	<p>практические занятия внеаудиторная самостоятельная работа индивидуальные задания</p>
<p>знания:</p> <ul style="list-style-type: none"> • 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. 	<p>практические занятия внеаудиторная самостоятельная работа индивидуальные задания зачет с оценкой</p>

- 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*

Оценка индивидуальных образовательных достижений по результатам текущего контроля производится в соответствии с универсальной шкалой

Процент результативности (правильных ответов)	Качественная оценка индивидуальных образовательных достижений	
	балл (отметка)	вербальный аналог
более 91%	5	отлично
от 80% до 90%	4	хорошо
от 61% до 79%	3	удовлетворительно
менее 60%	2	неудовлетворительно

Разработчик(и):

кафедра “Информационные технологии и гуманитарные науки” Ереванского филиала РЭУ им. Г.В. Плеханова): Нахатакян С.Х., к.т.н доцент

