

УТВЕРЖДАЮ
Первый проректор
С.В. Авдашкевич
«29» июля 2023

РАБОЧАЯ ПРОГРАММА

учебной дисциплины

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Специальность СПО:	<i>09.02.07 Информационные системы и программирование</i>
Форма обучения:	<i>очная</i>
Уровень образования, необходимый для приема на обучение по ППССЗ:	<i>среднее общее образование</i>

Санкт-Петербург
2023

СОДЕРЖАНИЕ

- 1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ**

ПРИЛОЖЕНИЕ 1

ПРИЛОЖЕНИЕ 2

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ.

1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС СПО по специальности 09.02.07 Информационные системы и программирование.

1.2. Место учебной дисциплины в структуре основной профессиональной образовательной программы:

учебная дисциплина «Основы алгоритмизации и программирование» принадлежит к общепрофессиональному циклу.

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

Процесс изучения дисциплины способствует формированию следующих компетенций:

Код	Содержание компетенции
ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;
ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.
ПК 5.4	Производить разработку модулей информационной системы в соответствии с техническим заданием
ПК 5.5	Осуществлять тестирование информационной системы на этапе опытной эксплуатации с фиксацией выявленных ошибок кодирования в разрабатываемых модулях информационной системы
ПК 9.3	Разрабатывать интерфейс пользователя веб-приложений в соответствии с техническим заданием
ПК 9.5	Производить тестирование разработанного веб-приложения

В результате изучения обязательной части учебного цикла обучающийся должен:

уметь:

- 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.

знать:

- 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, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.

В рамках рабочей программы используются активные и интерактивные формы проведения занятий: семинар-диалог, круглый стол, ролевые игры, проблемные лекции.

1.4. Количество часов на освоение рабочей программы учебной дисциплины:

максимальной учебной нагрузки обучающегося 186 часов, в том числе:

очная форма обучения

обязательной аудиторной учебной нагрузки обучающегося 160 часов;
самостоятельной работы обучающегося 8 часов;
экзамен 18 часов.

Распределение часов, добавленных за счет вариативной части ППСЗ, представлено в Приложении 1.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

2.1.1. Очная форма обучения

Вид учебной работы	Объем часов	Семестр	
		1	2
Максимальная учебная нагрузка обучающегося (всего)	186	72	114
Обязательная аудиторная учебная нагрузка обучающегося (всего)	160	68	92
В том числе:			
Лекционные занятия (ЛЗ)	80	34	46
Лабораторные работы (ЛР)			
Практические занятия, семинары (ПЗ)	80	34	46
Контрольные работы (КР)	-	+	-
Самостоятельная работа обучающегося (СР)	8	4	4
Консультации			
Форма промежуточной аттестации ¹	Э 18		Э 18

¹ Формы промежуточной аттестации (ДЗ – дифференцированный зачет, З – зачет, Э – экзамен) указываются в соответствии с учебным планом. Если в семестре не предусмотрена промежуточная аттестация, в соответствующей ячейке таблицы указывается «←» (другие формы контроля в таблице не указываются).

2.2. Тематический план и содержание учебной дисциплины.

Наименование разделов и тем	Содержание учебного материала, практические работы, самостоятельная работа обучающихся	Объем часов				Уровень освоения ³	
		Всего	в том числе				
			ЛЗ	ПЗ+ ЛР+ КР	СР		
Раздел 1. Введение в программирование		16	8	8			
Тема 1.1. Языки программирования	<p>Содержание учебного материала</p> <p>1. Развитие языков программирования.</p> <p>2. Обзор языков программирования. Области применения языков программирования. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы.</p> <p>3. Жизненный цикл программы.</p> <p>Программа. Программный продукт и его характеристики.</p> <p>4. Основные этапы решения задач на компьютере.</p> <p>Практические занятия</p> <p>Знакомство со средой программирования.</p> <p>Составление программ линейной структуры</p> <p>Составление программ разветвляющейся структуры</p> <p>Составление программ циклической структуры</p>		4			1,2	
			4				
Тема 1.2. Типы данных	<p>Содержание учебного материала</p> <p>1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.</p> <p>Практические занятия</p> <p>Обработка одномерных и двумерных массивов</p> <p>Обработка символьных строк</p>		4				
				4			
Раздел 2. Языки программирования		22	12	8	2		
Тема 2.1. Операторы языка программирования	<p>Содержание учебного материала</p> <p>1. Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор.</p> <p>2. Условный оператор. Оператор выбора.</p> <p>3. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы.</p> <p>4. Массивы. Двумерные массивы. Строки. Стандартные процедуры и функции для работы со строками.</p> <p>5. Структурированный тип данных – множество. Операции над множествами.</p> <p>6. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа</p> <p>Практические занятия</p> <p>Работа с текстовыми файлами</p> <p>Работа с двоичными файлами</p> <p>Самостоятельная работа</p> <p>Операции над множествами.</p>		12			2,3	
				8			2,3
					2		
Раздел 3. Типы программирования		34	12	20	2		
Тема 3.1. Процедуры и функции	<p>Содержание учебного материала</p> <p>1. Общие сведения о подпрограммах. Определение и вызов подпрограмм. Область видимости и время жизни переменной. Механизм передачи параметров. Организация функций.</p> <p>2. Рекурсия. Программирование рекурсивных алгоритмов.</p> <p>Практические занятия</p> <p>Организация функций</p> <p>Применение рекурсивных функций</p>		4				
				6			
Тема 3.2. Структуризация	<p>Содержание учебного материала</p> <p>1. Основы структурного программирования. Методы структурного</p>		4				

в программирова нии	программирования.					
	Практические занятия Программирование модуля			6		
	Самостоятельная работа Программирование рекурсивных алгоритмов				2	
Тема 3.3. Модульное программирова ние	Содержание учебного материала 1. Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы. 2. Стандартные модули.		4			
	Практические занятия Компиляция и компоновка программы			8		
Раздел 4. Основные конструкции языков программирования		8	6	2		
Тема 4.1 Указатели.	Содержание учебного материала 1. Указатели. Описание указателей. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных. 2. Структуры данных на основе указателей. 3. Задача о стеке.		6			
	Практические занятия Использование указателей для организации связанных списков. Создание и удаление динамических переменных			2		
Раздел 5. Принципы программирования		88	42	42	4	
Тема 5.1 Основные принципы объектно- ориентированно го программирова ния (ООП)	Содержание учебного материала 1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс. 2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм. 3. Классы объектов. Компоненты и их свойства. 4. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.		8			
	Практические занятия Структуры и объединения Классы и объекты			6		
	Самостоятельная работа					
Тема 5.2 Интегрированна я среда разработчика.	Содержание учебного материала 1. Требования к аппаратным и программным средствам интегрированной среды разработчика. 2. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов. 3. Панель компонентов и их свойства. Окно кода проекта. 4. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. 5. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. 6. Настройка среды и параметров проекта.		8			
	Практические занятия Изучение интегрированной среды разработчика. Приложения Windows Forms Создание проекта с использованием компонентов для работы с текстом Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени Создание проекта с использованием кнопочных компонентов Создание проекта с использованием списков выбора и полос прокрутки Создание проекта с использованием компонентов стандартных диалогов и системы меню			10		
	Самостоятельная работа Настройка среды и параметров проекта.				2	
Тема 5.3.	Содержание учебного материала		8			

Визуальное событийно-управляемое программирование	1. Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение. 2. Дополнительные элементы управления. Свойства компонентов. Виды свойств. Синтаксис определения свойств. Назначения свойств и их влияние на результат. Управление объектом через свойства. 3. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий.					
	Практические занятия Создание процедур обработки событий. Компиляция и запуск приложения					
Тема 5.4 Разработка оконного приложения Этапы разработки приложений	Содержание учебного материала 1. Разработка функционального интерфейса приложения. Создание интерфейса приложения. 2. Разработка функциональной схемы работы приложения. 3. Разработка игрового приложения. 4. Разработка приложения. 5. Проектирование объектно-ориентированного приложения. 6. Создание интерфейса пользователя. 7. Тестирование, отладка приложения.		10			
	Практические занятия Разработка оконного приложения с несколькими формами Разработка игрового приложения			10		
	Самостоятельная работа Тестирование, отладка приложения				2	
Тема 5.5 Иерархия классов.	Содержание учебного материала 1. Классы ООП: виды, назначение, свойства, методы, события. 2. Перегрузка методов. 3. Тестирование и отладка приложения. 4. Решение задач		8			
	Практические занятия Составление начальной иерархии и структуры классов			8		
Экзамен		18				
ВСЕГО		186	80	80	8	

³ 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 УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Для реализации учебной дисциплины требуются специальные помещения:

1. учебная аудитория для проведения лекционных, практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная специальной мебелью, компьютером с возможностью подключения к информационно-телекоммуникационной сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду, техническими средствами обучения, лицензионным программным обеспечением: операционная система MS Windows, пакет офисных программ Microsoft Office, антивирусная программа.

2. специализированная аудитория: лаборатория «Программирования и баз данных», оснащенная специальной мебелью, компьютерами с возможностью подключения к информационно-телекоммуникационной сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду, техническими средствами обучения и демонстрационными материалами, лицензионным программным обеспечением: ОС Microsoft Windows; Microsoft Office; Visual Studio Code; Notepad ++; Sublime Text 3; Microsoft Visual Studio; PyCharm; Python, виртуальный сервер.

3. помещение для самостоятельной работы, оснащенное специальной мебелью, компьютерной техникой с возможностью подключения к информационно-телекоммуникационной сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду, лицензионным программным обеспечением: операционная система MS Windows, пакет офисных программ MS Office, антивирусная программа.

3.2. Учебно-методическое и информационное обеспечение

Основная литература:

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, 2022. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/493261>
2. Кудрина, Е. В. Основы алгоритмизации и программирования на языке C#: учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва: Издательство Юрайт, 2022. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/494914>
3. Скорубский, В. И. Математическая логика: учебник и практикум для среднего профессионального образования / В. И. Скорубский, В. И. Поляков, А. Г. Зыков. — Москва: Издательство Юрайт, 2022. — 211 с. — (Профессиональное образование). — ISBN 978-5-534-11631-1. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/495977>

Дополнительная литература:

1. Судоплатов, С. В. Математика: математическая логика и теория алгоритмов: учебник и практикум для среднего профессионального образования / С. В. Судоплатов, Е. В. Овчинникова. — 5-е изд., стер. — Москва: Издательство Юрайт, 2022. — 255 с. — (Профессиональное образование). — ISBN 978-5-534-10930-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/495629>
2. Паронджанов, В. Д. Алгоритмические языки и программирование: ДРАКОН: учебное пособие для среднего профессионального образования /

В. Д. Паронджанов. — Москва : Издательство Юрайт, 2022. — 436 с. — (Профессиональное образование). — ISBN 978-5-534-14733-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/497329>

3. Черпаков, И. В. Основы программирования : учебник и практикум для среднего профессионального образования / И. В. Черпаков. — Москва : Издательство Юрайт, 2022. — 219 с. — (Профессиональное образование). — ISBN 978-5-9916-9984-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/491068>

Периодические издания

1. Прикладная информатика [Электронный ресурс] : журнал. — Режим доступа: https://www.elibrary.ru/title_about_new.asp?id=25599
2. Информационные системы и технологии [Электронный ресурс] : журнал. — Режим доступа: https://www.elibrary.ru/title_about.asp?id=28336
3. Международный журнал информационных технологий и энергоэффективности [Электронный ресурс]: журнал. — Режим доступа: https://elibrary.ru/title_about.asp?id=63461

Базы данных и информационные справочные системы:

1. ibooks.ru : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. <https://ibooks.ru/>
2. Электронно-библиотечная система СПбУТУиЭ : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. <https://libume.ru/>
3. Юрайт : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных <https://urait.ru/>
4. eLibrary.ru : научная электронная библиотека [Электронный ресурс] : профессиональная база данных. <https://www.elibrary.ru/>
5. КиберЛенинка : научная электронная библиотека [Электронный ресурс] : информационная справочная система. <http://cyberleninka.ru/>
6. Лань : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. <https://e.lanbook.com>
7. it-world.ru [Электронный ресурс] : информационная справочная система. <https://www.it-world.ru/>
8. Бизнес-информатика [Электронный ресурс] : информационная справочная система. <https://bijournal.hse.ru/>

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ 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>уметь:</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>знать:</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;• 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.	<p>Оценка результатов практических занятий, Устный фронтальный и индивидуальный опрос</p> <p>Оценка результатов тестирования, Самостоятельная работа, Подготовка и выступление с докладом, сообщением, презентацией</p> <p>Выполнение дополнительных заданий по собственной инициативе обучающихся</p>

Фонд оценочных средств представлен в Приложении 2.

Распределение часов вариативной части

Наименование разделов учебной дисциплины профессионального модуля (ПМ), междисциплинарных курсов (МДК) и тем	Требования к результатам освоения дисциплины, включая дополнительные требования	Распределение часов вариативной части
		Очная форма
Тема 2.1. Операторы языка программирования	<p>уметь:</p> <ul style="list-style-type: none"> • работать в среде программирования; • реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; <p>знать:</p> <ul style="list-style-type: none"> • основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; 	16
Тема 3.2. Структуризация в программировании	<p>уметь:</p> <ul style="list-style-type: none"> • разрабатывать алгоритмы для конкретных задач; • работать в среде программирования; <p>знать:</p> <ul style="list-style-type: none"> • основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; 	10
Тема 3.3. Модульное программирование	<p>уметь:</p> <ul style="list-style-type: none"> • разрабатывать алгоритмы для конкретных задач; • реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; <p>знать:</p> <ul style="list-style-type: none"> • основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; 	10
Тема 5.2 Интегрированная среда разработчика.	<p>уметь:</p> <ul style="list-style-type: none"> • разрабатывать алгоритмы для конкретных задач; • использовать программы для графического отображения алгоритмов; • определять сложность работы алгоритмов; <p>знать:</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; • подпрограммы, составление библиотек подпрограмм; • объектно-ориентированную модель 	10

	программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.	
Тема 5.3. Визуальное событийно- управляемое программирова- ние	<p>уметь:</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; <p>знать:</p> <ul style="list-style-type: none"> • 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. 	10
Тема 5.4 Разработка оконного приложения Этапы разработки приложений	<p>уметь:</p> <ul style="list-style-type: none"> • разрабатывать алгоритмы для конкретных задач; • оформлять код программы в соответствии со стандартом кодирования; • выполнять проверку, отладку кода программы. <p>знать:</p> <ul style="list-style-type: none"> • 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. 	26
Итого:		82