

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Катрич Ольга Владимировна  
Должность: директор  
Дата подписания: 21.09.2023 19:02:34  
Уникальный программный ключ:  
cfda5e8f32dda9141ed388a960d7348f631af091

Муниципальное государственное образовательное учреждение среднего профессионального образования  
«Новосибирский городской открытый колледж»  
(АОУ СПО «НГОК», НГОК)

Рассмотрено и принято  
на заседании Педагогического совета  
АОУ СПО «НГОК»  
Протокол № 4  
от «1» января 2023 года



О.В. Катрич  
1» января 2023 года

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ  
ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ  
специальности  
09.02.07 Информационные системы и программирование

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта 09.02.07. Информационные системы и программирование и примерной основной образовательной программы по специальности 09.02.07 «Информационные системы и программирование».

Организация-разработчик: АНО СПО «Новосибирский городской открытый колледж»

## СОДЕРЖАНИЕ

<b>1</b>	<b>ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>4</b>
<b>2</b>	<b>СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>6</b>
<b>3</b>	<b>УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>11</b>
<b>4</b>	<b>КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>12</b>

# 1 ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

## 1.1. Место дисциплины в структуре программы подготовки специалистов среднего звена:

Учебная дисциплина «Основы алгоритмизации и программирования» принадлежит к общепрофессиональному циклу.

## 1.2. Цель и планируемые результаты освоения дисциплины:

В результате освоения дисциплины 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.

В результате освоения дисциплины обучающийся *должен 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.

Реализация дисциплины направлена на формирование компетенций:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами;

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста;

ОК 09. Использовать информационные технологии в профессиональной деятельности;

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

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств

ПК 1.4 Выполнять тестирование программных модулей

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения;

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

### **1.3. Рекомендуемое количество часов на освоение программы дисциплины:**

максимальной учебной нагрузки обучающегося 152 часа, в том числе:

- обязательной аудиторной учебной нагрузки обучающегося- 132 часа;
- самостоятельной работы обучающегося- 10 часов;
- консультации- 2 часа;
- промежуточная аттестация- 8 часов.

## 2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### 2.1. Объем учебной дисциплины и виды учебной работы

<b>Вид учебной работы</b>	<b>Объем часов</b>
Объем учебной дисциплины (максимальный)	152
в том числе:	
Лекции, уроки	72
Практические занятия	60
Самостоятельная работа	10
Консультации	2
Промежуточная аттестация	8
<b>Промежуточная аттестация в форме экзамена</b>	

## 2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, практические занятия, самостоятельная работа обучающегося	Объем часов
1	2	3
<b>Раздел 1. Введение в программирование</b>		
1.1. Языки программирования	<b>Содержание учебного материала</b> Развитие языков программирования. Обзор языков программирования. 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.	<b>4</b>
	<b>Практические занятия</b> Знакомство со средой программирования.	<b>4</b>
	1.2. Типы данных	<b>Содержание учебного материала</b> Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.
	<b>Самостоятельная работа</b> Составление программ линейной структуры. Составление программ разветвляющейся структуры. Составление программ циклической структуры	<b>8</b>
<b>Раздел 2. Операторы языка программирования</b>		
2.1. Операторы языка программирования	<b>Содержание учебного материала</b> Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор. Условный оператор. Оператор выбора. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы. Массивы. Двумерные массивы. Строки. Стандартные процедуры и функции для работы со строками. Структурированный тип данных – множество. Операции над множествами. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа.	<b>4</b>

	<b>Практические занятия</b>	<b>4</b>
	Операции над множествами. Файлы прямого доступа.	
<b>Раздел 3.</b>		
3.1. Процедуры и функции	<b>Содержание учебного материала</b>	<b>4</b>
	Общие сведения о подпрограммах. Определение и вызов подпрограмм. Область видимости и время жизни переменной. Механизм передачи параметров. Организация функций. Рекурсия. Программирование рекурсивных алгоритмов.	
	<b>Практические занятия</b>	<b>4</b>
	Организация функций. Рекурсия. Программирование рекурсивных алгоритмов.	
3.2. Структуризация в программировании	<b>Содержание учебного материала</b>	<b>4</b>
	Основы структурного программирования. Методы структурного программирования.	
	<b>Практические занятия</b>	<b>4</b>
	Методы структурного программирования.	
3.3. Модульное программирование	<b>Содержание учебного материала</b>	<b>4</b>
	Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы. Стандартные модули.	
	<b>Практические занятия</b>	<b>4</b>
	Программирование модуля. Создание библиотеки подпрограмм.	
<b>Раздел 4. Основы конструкции языков программирования</b>		
4.1. Указатели	<b>Содержание учебного материала</b>	<b>4</b>
	Указатели. Описание указателей. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных. Структуры данных на основе указателей. Задача о стеке.	
	<b>Практические занятия</b>	<b>4</b>
	Использование указателей для организации связанных списков.	
<b>Раздел 5.</b>		



5.1. Основные принципы объектно-ориентированного программирования	<b>Содержание учебного материала</b>	4
	История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс. Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Классы объектов. Компоненты и их свойства. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.	
	<b>Самостоятельная работа</b>	2
5.2. Интегрированная среда разработчика	<b>Содержание учебного материала</b>	8
	Требования к аппаратным и программным средствам интегрированной среды разработчика. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. Настройка среды и параметров проекта.	
	<b>Практические занятия</b>	4
5.3. Визуальное событийно-управляемое программирование	<b>Содержание учебного материала</b>	8
	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.	
	<b>Практические занятия</b>	4
	События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий.	

	Создание проекта с использованием кнопочных компонентов. Создание проекта с использованием компонентов стандартных диалогов и системы меню.	
5.4. Разработка оконного приложения	<b>Содержание учебного материала</b> Разработка функционального интерфейса приложения. Создание интерфейса приложения. Разработка функциональной схемы работы приложения. Разработка игрового приложения.	<b>8</b>
	<b>Практические занятия</b> Разработка функциональной схемы работы приложения. Разработка оконного приложения с несколькими формами. Разработка игрового приложения.	<b>8</b>
5.5. Этапы разработки приложений	<b>Содержание учебного материала</b> Разработка приложения. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя. Тестирование, отладка приложения.	<b>8</b>
	<b>Практические занятия</b> Создание процедур обработки событий. Компиляция и запуск приложения. Разработка интерфейса приложения. Тестирование, отладка приложения.	<b>8</b>
5.6. Иерархия классов	<b>Содержание учебного материала</b> Классы ООП: виды, назначение, свойства, методы, события. Перегрузка методов. Тестирование и отладка приложения. Решение задач	<b>8</b>
	<b>Практические занятия</b> Классы ООП: виды, назначение, свойства, методы, события. Объявления класса. Создание наследованного класса.	<b>8</b>
	<b>Консультации</b>	<b>2</b>
	<b>Промежуточная аттестация</b>	<b>8</b>
	<b>ВСЕГО</b>	<b>152</b>

### **3 УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

**3.1. Для реализации программы учебной дисциплины должны быть предусмотрены следующие специальные помещения:**

Лаборатория «**Программирование баз данных**», оснащенная необходимым для реализации программы учебной дисциплины оборудованием, приведенным в п 6.1.2.1 примерной программы по данной специальности.

**3.2. Информационное обеспечение обучения. Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы.**

1. Кудинов, Ю. И. Основы алгоритмизации и программирования : учебное пособие для СПО / Ю. И. Кудинов, А. Ю. Келина. — 2-е изд. — Липецк, Саратов : Липецкий государственный технический университет, Профобразование, 2020. — 71 с. — ISBN 978-5-88247-956-4, 978-5-4488-0757-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/92834.html>

2. Тюльпинова, Н. В. Алгоритмизация и программирование : учебное пособие / Н. В. Тюльпинова. — Саратов : Вузовское образование, 2019. — 200 с. — ISBN 978-5-4487-0470-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/80539.html>

3. Чурина, Т. Г. Основы алгоритмизации и программирования : учебное пособие для СПО / Т. Г. Чурина, Т. В. Нестеренко. — Саратов, Москва : Профобразование, Ай Пи Ар Медиа, 2020. — 214 с. — ISBN 978-5-4488-0802-9, 978-5-4497-0465-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/96017.html>

## 4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

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

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<b>освоенные умения</b>	
<ul style="list-style-type: none"> <li>- разрабатывать алгоритмы для конкретных задач;</li> <li>- использовать программы для графического отображения алгоритмов;</li> <li>- определять сложность работы алгоритмов;</li> <li>- работать в среде программирования;</li> <li>- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;</li> <li>- оформлять код программы в соответствии со стандартом кодирования;</li> <li>- выполнять проверку, отладку кода программы.</li> </ul>	<ul style="list-style-type: none"> <li>Самостоятельная работа;</li> <li>Наблюдение за выполнением практических заданий;</li> <li>Оценка выполнения практического задания;</li> <li>Выступление с докладом, сообщением, презентацией;</li> <li>Решение ситуационных задач</li> </ul>
<b>усвоенные знания</b>	
<ul style="list-style-type: none"> <li>- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;</li> <li>- эволюцию языков программирования, их классификацию, понятие системы программирования;</li> <li>- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;</li> <li>- подпрограммы, составление библиотек подпрограмм;</li> <li>- 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.</li> </ul>	<ul style="list-style-type: none"> <li>Оценка результатов выполнения практической работы;</li> <li>Экспертное наблюдение за ходом выполнения практической работы</li> </ul>
<b>Промежуточная аттестация- экзамен</b>	