

государственное автономное профессиональное образовательное
учреждение Самарской области
«Новокуйбышевский гуманитарно-технологический колледж»

УТВЕРЖДЕНО
Приказ директора
ГАПОУ «НГТК»
От 06.02.2024 г. № 30.1-К

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ОП.04 Основы алгоритмизации и программирования

для специальности:

09.02.07 Информационные системы и программирование

профиль подготовки: технологический

на базе основного общего образования

СОДЕРЖАНИЕ

| | |
|---|----|
| 1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ | 4 |
| 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ | 6 |
| 3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ..... | 10 |
| 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ | 12 |

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1 Область применения рабочей программы

Рабочая программа учебной дисциплины ОП.04 Основы алгоритмизации и программирования является частью программы подготовки специалистов среднего звена по специальности 09.02.07 Информационные системы и программирование.

Рабочая программа учебной дисциплины может быть использована для студентов специальности 09.02.07 Информационные системы и программирование, для курсов повышения квалификации и переподготовки.

1.2 Место учебной дисциплины в структуре основной профессиональной образовательной программы

Общепрофессиональная дисциплина профессионального цикла.

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

В результате освоения учебной 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 учебной 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.

Процесс изучения дисциплины направлен на формирование общих компетенций (ОК) и профессиональных компетенций (ПК)

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. Эффективно взаимодействовать и работать в коллективе и команде.

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.

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.

1.4 Количество часов на освоение рабочей программы учебной дисциплины

– обязательной аудиторной учебной нагрузки обучающегося 156 часов.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

| Вид учебной работы | Объем часов |
|---|--------------------|
| Обязательная аудиторная учебная нагрузка (всего) | 156 |
| в том числе: | |
| практические занятия | 100 |
| | |
| Итоговая аттестация в форме дифференцированного зачета | |

2.2 Содержание учебной дисциплины ОП.04 Основы алгоритмизации и программирования

| Наименование тем учебной дисциплины | Содержание учебного материала, практические занятия, самостоятельная работа обучающихся, курсовая работа (проект) | Объем часов | Код образовательного результата ФГОС СПО | |
|--|--|-------------|--|----|
| 1 | 2 | 3 | 4 | |
| Тема 1. Введение в алгоритмизацию и программирование на языке C# | Содержание | 8 | | |
| | 1. История развития языков программирования | | | |
| | 2. Основные понятия и термины | | | |
| | 3. Языки программирования и их классификация | | | |
| | 4. Преимущества и особенности языка C# | | | |
| Тема 2. Изучение синтаксиса и основных конструкций языка C# | Содержание | 28 | ПК 2.4 ПК 2.5 ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 | |
| | 1. Переменные, типы данных и операции | | | |
| | 2. Условные операторы и циклы | | | |
| | 3. Массивы и коллекции | 22 | | |
| | Практическое занятие 1,2,3,4,5,6,7,8,9,10,11 | | | |
| | 1. Простые вычисления: сложение, вычитание, умножение и деление чисел | | | |
| | 2. Решение квадратного уравнения | | | |
| | 3. Обработка одномерного массива: поиск максимального и минимального элемента, сортировка массива | | | |
| | 4. Обработка двумерного массива: заполнение случайными числами, нахождение суммы элементов | | | |
| | 5. Создание простого калькулятора с функциями сложения, вычитания, умножения и деления | | | |
| | 6. Создание простой программы для решения линейного уравнения с одной переменной | | | |
| | 7. Программирование простого текстового редактора с функциями создания, открытия, сохранения и редактирования текста | | | |
| | 8. Создание простого графического приложения, выводящего на экран геометрические фигуры (например, окружность, прямоугольник, линию) | | | |
| | 9. Программирование калькулятора для решения квадратных уравнений | | | |
| | 10. Создание программы для решения системы линейных уравнений методом Гаусса | | | |
| 11. Написание программы для работы с матрицами: сложение, умножение на число, транспонирование | | | | |
| Тема 3. Основы алгоритмики | Содержание | 20 | | |
| | 1. Алгоритмы и их свойства | | | |
| | 2. Классификация алгоритмов | | | |
| | Практические занятия 12,13,14,15,16,17,18,19 | | | 16 |
| | 1. Разработка алгоритма нахождения суммы элементов массива | | | |
| | 2. Разработка алгоритма сортировки массива методом пузырька | | | |
| | 3. Разработка алгоритма поиска максимального элемента в массиве | | | |

| | | | | |
|---|--|--|-----------|--|
| | 4. | Разработка алгоритма решения квадратного уравнения | | |
| | 5. | Разработка алгоритма бинарного поиска элемента в отсортированном массиве | | |
| | 6. | Разработка алгоритма быстрой сортировки массива | | |
| | 7. | Разработка алгоритма подсчета количества элементов в массиве, удовлетворяющих некоторому условию | | |
| | 8. | Разработка алгоритма, который находит все простые числа в заданном диапазоне | | |
| Тема 4. Объектно-ориентированное программирование в C# | Содержание | | 36 | |
| | 1. | Классы и объекты в C# | | |
| | 2. | Инкапсуляция: основные принципы и применение на практике | | |
| | 3. | Свойства и методы классов: определение, использование и модификаторы доступа | | |
| | 4. | Конструкторы и деструкторы классов: назначение, использование и параметры | | |
| | 5. | Статические элементы классов: свойства, методы и поля | | |
| | 6. | Наследование: определение и принципы работы | | |
| | 7. | Полиморфизм: основные понятия и использование на примерах | | |
| | 8. | Абстрактные классы и интерфейсы: назначение и использование | | |
| | 9. | Вложенные и внутренние классы: использование и особенности | | |
| | Практические занятия 20,21,22,23,24,25,26,27,28 | | 18 | |
| 1. | Создание класса «Автомобиль» с определенными методами | | | |
| 2. | Создание класса «Велосипед» с определенными методами | | | |
| 3. | Создание класса «Кошка» с определенными методами | | | |
| 4. | Создание абстрактного класса «Животное» с определенными методами | | | |
| 5. | Создание классов «Книга» и «Журнал» с определенными методами | | | |
| 6. | Создание интерфейса «ICalculator» и классов, с различными реализациями методов | | | |
| 7. | Создание базового класса «Shape» с определенными методами и производными от него классами | | | |
| 8. | Реализация паттерна «Factory Method» с абстрактными и конкретными классами, каждый из которых имеет свой метод | | | |
| 9. | Реализация паттернов «Strategy» и «State» с абстрактными и конкретными классами и состояниями | | | |
| Тема 5. Работа с файлами и потоками ввода-вывода | Содержание | | 20 | |
| | 1. | Чтение и запись данных | | |
| | 2. | Работа с ошибками ввода-вывода | | |
| | Практические занятия 29,30,31,32,33,34,35,36 | | 16 | |
| 1. | Открытие и закрытие файла. Чтение и запись данных в файл. Создание и удаление файлов | | | |
| 2. | Простейшие операции с каталогами (создание, удаление, переименование) | | | |
| 3. | Использование потоков ввода-вывода для чтения и записи данных | | | |
| 4. | Работа с ошибками ввода-вывода, обработка исключений | | | |
| 5. | Использование буферизованного и небуферизованного ввода-вывода | | | |
| 6. | Использование асинхронного ввода-вывода и многопоточности | | | |

| | | | | |
|---|--|--|------------|--|
| | 7. | Сериализация и десериализация данных с использованием XML, JSON, бинарных форматов | | |
| | 8. | Работа с ZIP-архивами | | |
| Тема 6. Исключения и обработка ошибок | Содержание | | 14 | |
| | 1. | Исключения в C# | | |
| | 2. | Обработка исключений | | |
| | Практические занятия 37,38,39,40,41 | | 10 | |
| | 1. | Использование ключевых слов throw и try-catch-finally для обработки исключений | | |
| | 2. | Использование блоков try-catch для обработки ошибок времени выполнения | | |
| 3. | Создание пользовательских исключений и их обработка | | | |
| 4. | Использование исключений для управления потоком выполнения программы | | | |
| | 5. | Отладка исключений с помощью инструментов Visual Studio | | |
| Тема 7. Разработка программ и проектов | Содержание | | 28 | |
| | 1. | Создание проектов в Visual Studio | | |
| | 2. | Использование NuGet для установки и управления пакетами и библиотеками | | |
| | 3. | Использование основных стандартных библиотек .NET | | |
| | 4. | Использование сторонних библиотек и пакетов | | |
| | 5. | Создание многофайловых проектов с использованием папок и подпапок | | |
| | Практические занятия 42,43,44,45,46,47,48,49,50 | | 18 | |
| | 1. | Разработка приложения Windows Forms на C# и реализация работы с формами, элементами управления и событиями | | |
| | 2. | Разработка программы, которая использует классы, структуры, интерфейсы и перечисления в C# | | |
| | 3. | Разработка программы на C#, которая демонстрирует использование обобщений | | |
| | 4. | Разработка программы на C# для работы со строками с использованием регулярных выражений | | |
| | 5. | Разработка консольного приложения на C# и использовать в нем многопоточность с использованием классов Thread и Tasks | | |
| | 6. | Разработка приложения Windows Forms на C# и реализовать в нем работу с формами, элементами управления и событиями | | |
| | 7. | Разработка веб-приложения ASP.NET Core и использовать в нем MVC, Razor Pages, Entity Framework Core и SQL Server | | |
| | 8. | Разработка индивидуального проекта на языке программирования C# | | |
| 9. | Разработка индивидуального проекта на языке программирования C# | | | |
| Дифференцированный зачет | | | 2 | |
| Всего | | | 156 | |

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1 Требования к минимальному материально-техническому обеспечению

Реализация учебной дисциплины требует наличия учебного кабинета и лаборатории.

Оборудование учебного кабинета: стандартное оборудование рабочих мест преподавателя и студента.

Технические средства обучения: персональный компьютер, мультимедиа проектор, экран.

Оборудование лаборатории и рабочих мест лаборатории:

Персональный компьютер и периферийные устройства (принтер, картриджи, сканер, винчестер, CD и DVD-ROM и др.) для проведения практических работ, телекоммуникационные средства.

3.2 Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Зыков, С. В. Программирование: учебник и практикум для академического бакалавриата / С. В. Зыков. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2023.

2. Казанский, А. А. Программирование на Visual C#: учебное пособие для среднего профессионального образования / А. А. Казанский. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2023.

3. 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, 2023.

4. Подбельский, В. В. Программирование. Базовый курс C#: учебник для среднего профессионального образования / В. В. Подбельский. — Москва: Издательство Юрайт, 2023.

Дополнительные источники:

1. Программирование: математическая логика: учебное пособие для вузов / М. В. Швецкий, М. В. Демидов, А. В. Голанова, И. А. Кудрявцева. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2023.

2. Трофимов, В. В. Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — 4-е изд. — Москва: Издательство Юрайт, 2023.

3. Черпаков, И. В. Основы программирования: учебник и практикум для среднего профессионального образования / И. В. Черпаков. — Москва: Издательство Юрайт, 2023.

4. Черпаков, И. В. Основы программирования: учебник и практикум для вузов / И. В. Черпаков. — Москва: Издательство Юрайт, 2023.

Интернет-ресурсы:

1. <http://www.intuit.ru> Интернет-университет информационных технологий.
2. <http://www.iteach.ru> программа Intel «Обучение для будущего».
3. <http://alglib.sources.ru> Проект Alglib: библиотека алгоритмов.
4. <http://algotlist.manual.ru> – Проект Alglib: алгоритмы и методы.
5. <http://www.rusedu.info> – Информационные технологии в образовании.
6. <http://www.ipospb.ru/journal> – Компьютерные инструменты в образовании.
7. <http://conf.pskovedu.ru> – Интеграция информационных систем в образовании.
8. <http://www.konkurskit.ru> – КИТ – компьютеры, информатика, технологии.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

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

| Результаты обучения (освоенные умения, усвоенные знания) | Критерии оценки | Методы оценки |
|--|---|--|
| <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>Оценка «отлично» выставляется обучающемуся, если он глубоко и прочно усвоил программный материал курса, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами и вопросами, не затрудняется с ответами при видоизменении заданий, правильно обосновывает принятые решения, владеет разносторонними дискуссионными навыками и приемами, активно проявляет себя в групповой работе;</p> <p>Оценка «хорошо» выставляется обучающемуся, если он твердо знает материал курса, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении дискуссионных вопросов и задач, владеет необходимыми навыками и приемами их выполнения, способен проявлять себя в групповой работе;</p> | <p>Текущий контроль: - опрос устный; - тестирование; - выполнение письменной работы; - выполнение практической работы</p> <p>Оценка результатов выполнения самостоятельной работы.</p> <p>Дифференцированный зачет</p> |
| <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 | <p>Оценка «удовлетворительно» выставляется обучающемуся, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно</p> | |

| | | |
|---|---|--|
| <p>программирования;</p> <ul style="list-style-type: none"> – основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; – подпрограммы, составление библиотек подпрограмм. | <p>правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических задач, не активен в групповой работе;</p> <p>Оценка «неудовлетворительно» выставляется обучающемуся, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями решает практические задачи или не справляется с ними самостоятельно, не принимает участие в групповой работе.</p> | |
|---|---|--|