1. **Паспорт Образовательной программы**

**«**Программирование на С/С++ **»**

|  |  |
| --- | --- |
| **Версия программы** | **1** |
| **Дата Версии** | 06**.**10**.**2020 |

1. **Сведения о Провайдере**

|  |  |  |
| --- | --- | --- |
| 1.1 | Провайдер | ФГБОУ ВО "Чувашский государственный университет имени И.Н. Ульянова" |
| 1.2 | Логотип образовательной организации | tovarnznak-1.jpeg |
| 1.3 | Провайдер ИНН | 2129009412 |
| 1.4 | Ответственный за программу ФИО | Ильина Лариса Алексеевна |
| 1.5 | Ответственный должность | Старший преподаватель |
| 1.6 | Ответственный Телефон | +79023285579 |
| 1.7 | Ответственный Е-mail | larisai2009@gmail.com |

1. **Основные Данные**

|  |  |  |
| --- | --- | --- |
| **№** | **Название** | **Описание** |
| 2.1 | Название программы | Программирование на С/С++ |
| 2.2 | Ссылка на страницу программы | https://cdochgu.ru/povyshenie-kvalifikatsii-i-professionalnaya-perepodgotovka/item/programmirovanie-na-s-s.html |
| 2.3 | Формат обучения | Онлайн |
|  | Подтверждение от ОО наличия возможности реализации образовательной программы с применением электронного обучения и (или) дистанционных образовательных технологий с возможностью передачи данных в форме элементов цифрового следа | Имеется возможность реализации образовательной программы с применением электронного обучения и (или) дистанционных образовательных технологий на базе вебинарной платформы университета, системы управления курсами Moodle, автоматизированных тестирующих систем |
| 2.4 | Уровень сложности | Базовый |
| 2.5 | Количество академических часов | **72** |
|  | Практикоориентированный характер образовательной программы: не менее 50 % трудоёмкости учебной деятельности отведено практическим занятиям и (или) выполнению практических заданий в режиме самостоятельной работы (кол-во академических часов) | 48 академических часов отведено на практические занятия и выполнение практических занятий в режиме самостоятельной работы (написание программного кода) |
| 2.6 | Стоимость обучения одного обучающегося по образовательной программе, а также предоставление ссылок на 3 (три) аналогичные образовательные программы иных организаций, осуществляющих обучение, для оценки объективности стоимости или обоснование уникальности представленной образовательной программы в случае отсутствия аналогичных образовательных программ на рынке образовательных услуг | 20000  Аналогичные образовательные программы:  [http://edu.bmstu.ru/napravleniya-obucheniya/programmirovanie/cpp - 25290](http://edu.bmstu.ru/napravleniya-obucheniya/programmirovanie/cpp%20%20-%2025290) (объем программы – 48 ак.ч.)  <https://www.avalon.ru/Courses/Vendors/CPP/> 21100 (объем программы –44 ак.ч.)  [https://www.specialist.ru/course/sipp](https://www.specialist.ru/course/sipp%20)  27990 (объем программы – 40 ак.ч.) |
| 2.7 | Минимальное количество человек на курсе | 5 |
| 2.8 | Максимальное количество человек на курсе | 40 |
| 2.9 | Данные о количестве слушателей, ранее успешно прошедших обучение по образовательной программе | 20 |
| 2.10 | Формы аттестации | Текущий контроль, промежуточная аттестация по модулю, итоговая аттестация в форме зачета |
|  | Указание на область реализации компетенций цифровой экономики, к которой в большей степени относится образовательная программа, в соответствии с Перечнем областей | Программирование и создание ИТ-продуктов |

1. **Аннотация программы**

Наиболее полное и содержательное описание программы, которое включает:

1) общую характеристику компетенций, качественное изменение которых осуществляется в результате обучения или которые формируются в результате освоения образовательной программы;

2) описание требований и рекомендаций для обучения по образовательной программе;

3) краткое описание результатов обучения в свободной форме, а также описание востребованности результатов обучения в профессиональной деятельности.

Ограничение по размеру: не менее 1000 символов -?

1) Компетенция формируемая в процессе обучения по программе повышения квалификации соответствует одной из обобщенных трудовых функций профессионального стандарта «Программист» (утв. п[риказом Минтруда России №679н от 18 ноября 2013 г.](https://mintrud.gov.ru/docs/mintrud/orders/138)) и направлена на формирование у слушателя способности разрабатывать и осуществлять отладку программного кода, в том числе осуществлять формализацию и алгоритмизацию поставленных задач; разрабатывать программный код с использованием языков программирования, определять и манипулировать данными; проверять и отлаживать программный код. В результате освоения программы повышения квалификации обучающиеся должны:

Знать:

Языки формализации функциональных спецификаций;

Методы и приемы алгоритмизации поставленных задач;

Алгоритмы решения типовых задач, области и способы их применения;

Синтаксис языков программирования С/С++, их особенности, стандартные библиотеки;

Типы и форматы сообщений об ошибках, предупреждений;

Современные компиляторы, отладчики и оптимизаторы программного кода.

Уметь

Использовать методы и приемы алгоритмизации поставленных задач;

Применять стандартные алгоритмы;

Разрабатывать программный код с использованием языков программирования С/С++;

Применять методы и приемы отладки программного кода;

Владеть навыками:

Формализации и алгоритмизации поставленных задач;

Разработки программ с использованием языков программирования высокого уровня С/С++;

Применения методов и приемов отладки и оптимизации программного кода;

Использования интегрированных средств разработки для создания программных продуктов.

2) Слушатели программы повышения квалификации должны иметь среднее профессиональное и/или высшее образование по направлению подготовки (специальности) в области математических и естественных наук, инженерного дела, технологий и технических наук и/или иметь опыт работы в сфере информатизации и автоматизации бизнес-процессов предприятий и организаций. Программа будет полезна для педагогов информатики и информационно-коммуникационных технологий. Использование дистанционных технологий и электронного обучения в процессе изучения материала программы требует наличия у слушателей ПЭВМ с доступом в Интернет, оборудованных видеокамерой, микрофоном и установленными средами программирования на языках С/С++ (свободные и открытые лицензии)

3) Программа повышения квалификации «Программирование на С/С++» рассчитана на слушателей, которые хотят приобрести (развить) компетенцию, связанную со способностью разрабатывать и осуществлять отладку программного кода на языках программирования С/С++. Программа имеет базовый уровень сложности. Содержание программы предусматривает освоение материала по таким темам как: организация взаимодействия пользователя с ЭВМ; общая характеристика языков программирования и области их применения; основные этапы разработки приложений; способы описания алгоритмов; тестирование и отладка; элементы языка; структура программы; операторы; массивы; функции; указатели; структуры; файлы; объектно-ориентированное программирование в С++; библиотека стандартных шаблонов. Языки программирования С/С++ широко применяются при разработке информационных и автоматизированных систем в различных отраслях экономики. Полученные в процессе обучения знания позволят освоить в дальнейшем такие современные языки программирования, как C#, Java. Программа будет полезна для педагогов информатики и информационно-коммуникационных технологий.

Востребованность программы подтверждается результатами анализа состояния и перспективами развития отрасли информационных технологий на период до 2025 года, представленными в распоряжении Правительства РФ от 01 ноября 2013 года №2036-р, в том числе острым дефицитом кадров, обладающих навыками программирования, недостаточным уровнем подготовки уже работающих специалистов, чей должностной функционал в условиях цифровизации производства и интеллектуализации выпускаемой продукции требует расширения за счет применения навыков программирования.

1. ШАБЛОН ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ (ДПО)

Титульный лист программы

Название организации

Дополнительная профессиональная программа повышения квалификации

« ФГБОУ ВО "Чувашский государственный университет имени И.Н. Ульянова"

"Программирование на С/С++" »

72 час.

**ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ**

**1.Цель программы**

Целью реализации программы "Программирование на С/С++" является формирование (развитие) у обучающегося компетенции по направлению цифровой экономики "Программирование и создание ИТ-продуктов" необходимой для осуществления профессиональной деятельности и (или) повышение профессионального уровня в рамках имеющейся квалификации специалистов, работающих в сфере информационных технологий и разработки программного обеспечения.

**2.Планируемые результаты обучения:**

2.1.Знание (осведомленность в областях)

2.1.1. Языков формализации функциональных спецификаций;

2.1.2. Методов и приемов алгоритмизации поставленных задач;

2.1.3. Алгоритмов решения типовых задач, областей и способов их применения;

2.1.4. Синтаксиса языков программирования С/С++, их особенностей, стандартных библиотек;

2.1.5. Типов и форматов сообщений об ошибках, предупреждений;

2.1.6. Современных компиляторов, отладчиков и оптимизаторов программного кода.

2.2. Умение (способность к деятельности)

2.2.1. Использовать методы и приемы алгоритмизации поставленных задач;

2.2.2. Применять стандартные алгоритмы;

2.2.3. Разрабатывать программный код с использованием языков программирования С/С++;

2.2.4. Применять методы и приемы отладки программного кода

2.3.Навыки (использование конкретных инструментов)

2.3.1. Формализации и алгоритмизации поставленных задач;

2.3.2. Разработки программ с использованием языков программирования высокого уровня С/С++;

2.3.3. Применения методов и приемов отладки и оптимизации программного кода;

2.3.4. Использования интегрированных средств разработки для создания программных продуктов

**3.Категория слушателей** (возможно заполнение не всех полей)

* 1. Среднее профессиональное и/или высшее образование
  2. Квалификация по направлению подготовки (специальности) в области математических и естественных наук, инженерного дела, технологий и технических наук, в области информационно-коммуникационных технологий, преподаватели информатики и информационно-коммуникационных технологий.
  3. Опыт работы в сфере информационно-коммуникационных технологий, информатизации и автоматизации бизнес-процессов предприятий и организаций.
  4. Цифровая грамотность, первичные знания и умения в области алгоритмизации и программирования

**4.Учебный план программы «…..наименование программы….»**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№ п/п** | **Модуль** | **Всего, час** | **Виды учебных занятий** | | |
| **лекции** | **практические занятия** | **самостоятельная работа** |
|  | Технология разработки алгоритмов и приложений | 11 | 4 | 4 | 3 |
|  | Программирование на языках высокого уровня С/С++ | 58 | 20 | 20 | 18 |
| **Итоговая аттестация** | |  | **Указывается вид (экзамен, зачёт, реферат и т.д.)** | | |
| Итоговая аттестация | | 3 | Зачет | | |

**5.Календарный план-график реализации образовательной** программы

(дата начала обучения – дата завершения обучения) в текущем календарном году, указания на периодичность набора групп (не менее 1 группы в месяц)

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Наименование учебных модулей** | **Трудоёмкость (час)** | **Сроки обучения** |
| 1 | Технология разработки алгоритмов и приложений | 11 | 02.11.2020-03.11.2020 |
| 2 | Программирование на языках высокого уровня С/С++ | 58 | 05.11.2020-  13.11.2020 |
|  | Итоговая аттестация | 3 | 14.11.2020 |
| **Всего:** | | 72 | 02.11.2020-14.11.2020 |

**6.Учебно-тематический план программы «**  Программирование на С/С++ **»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№ п/п** | **Модуль / Тема** | **Всего, час** | **Виды учебных занятий** | | | **Формы контроля** |
| **лекции** | **практические занятия** | **самостоятельная работа** |
| 1 | Технология разработки алгоритмов и приложений | 11 | 4 | 4 | 3 | Текущий контроль в форме тестирования знаний, тестирование разработанного программного кода |
| 1.1 | Организация взаимодействия пользователя с ЭВМ | 1 | 1 | 0 | 0 |  |
| 1.2 | Общая характеристика языков программирования, области их применения | 1 | 1 | 0 | 0 |  |
| 1.3 | Основные этапы разработки приложений. Способы описания алгоритмов | 3 | 1 | 1 | 1 |  |
| 1.4 | Тестирование и отладка | 6 | 1 | 3 | 2 |  |
| 2 | Программирование на языках высокого уровня С/С++ | 58 | 20 | 20 | 18 | Текущий контроль в форме тестирования знаний, тестирование разработанного программного кода |
| 2.1 | Элементы языка. Структура программы. Операторы | 10 | 4 | 4 | 2 |  |
| 2.2 | Массивы. Функции. Указатели. Структуры. Файлы | 16 | 6 | 5 | 5 |  |
| 2.3 | Основы объектно-ориентированного программирования в С++ | 21 | 6 | 7 | 8 |  |
| 2.4 | Библиотека стандартных шаблонов | 11 | 4 | 4 | 3 |  |
|  | Итоговая аттестация | 3 |  | 3 |  | Зачет |
|  | Итого | 72 | 24 | 27 | 21 |  |

**7. Учебная (рабочая) программа повышения квалификации «** Программирование на С/С++ **»**

Модуль 1. Технология разработки алгоритмов и приложений **(**11 **час.)**

Тема 1.1 Организация взаимодействия пользователя с ЭВМ

Назначение, состав и структура программного обеспечения. Организация взаимодействия пользователя с ЭВМ.

Тема 1.2. Общая характеристика языков программирования, области их применения

Общая характеристика языков программирования, области их применения. Компиляторы и интерпретаторы.

Тема 1.3. Основные этапы разработки приложений. Способы описания алгоритмов

Технология разработки алгоритмов и приложений. Основные этапы разработки приложений. Определение алгоритма. Свойства алгоритма. Способы описания алгоритмов: словесный, схемный, с помощью псевдокода или языка программирования. Методы разработки алгоритмов и программ: нисходящее, восходящее. Модульное представление программ.

Тема 1.4. Тестирование и отладка

Методы тестирования. Типы ошибок. Способы и средства обнаружения и локализации синтаксических и логических ошибок. Организация отладки и тестирования приложений.

Модуль 2. Программирование на языках высокого уровня С/С++ ( 58 час.)

Тема 2.1. Элементы языка. Структура программы. Операторы

Алфавит, лексемы, идентификаторы, ключевые слова. Переменные и константы. Знаки операций. Приоритеты операций. Условный оператор и переключатель. Структура программы. Определение констант и типов данных. Приведение типов и функции преобразования типов. Операторы. Цикл с предусловием, цикл с постусловием и цикл с параметром. Операторы break и continue. Форматированный ввод и вывод данных. Потоковый ввод и вывод данных.

Тема 2.2. Массивы. Функции. Указатели. Структуры. Символы и строки. Файлы

Одномерные и многомерные массивы. Инициализация и ввод массивов. Стандартные алгоритмы обработки массивов. Выделение памяти. Динамические массивы.

Объявление функции. Механизмы передачи параметров. Локальные и глобальные параметры. Область видимости и время жизни переменной. Рекурсивные функции. Передача параметров. Функции с переменным количеством параметров. Перегрузка и шаблоны функций.

Адреса, указатели и ссылки. Указатели в параметрах функций. Указатели на функции, массивы указателей на функции.

Структурные типы и структуры. Объявление структурного типа, структуры, указателя на структуру. Доступ к компонентам структур. Массивы структур. Структуры и функции. Способы представления текстов. Символы и строки. Встроенные подпрограммы обработки строк. Текстовые, типизированные и двоичные файлы. Прямой и последовательный доступ.

Тема  2.3. Основы объектно-ориентированного программирования в С++

Концепции объектно-ориентированного программирования. Тип данных класс. Определение класса и его членов. Конструкторы и деструкторы. Объекты. Массивы объектов. Инкапсуляция. Закрытые, открытые и защищенные члены класса. Обеспечение доступа. Дружественные функции. Наследование. Организация наследования классов. Иерархия и множественное наследование. Конструкторы, деструкторы и наследование. Полиморфизм. Перегрузка методов в классе. Конструкторы копий. Перегрузка операторов. Перегрузка ввода-вывода. Виртуальные методы. Виртуальные классы и виртуальные методы.

Тема 2.4. Библиотека стандартных шаблонов

Обзор библиотеки STL. Контейнеры. Итераторы. Алгоритмы. Распределители памяти. Предикаты. Функции сравнения. Контейнерные классы. Класс String. Шаблонная спецификация класса. Конструкторы класса. Операторы. Функции члены. Поиск в строке. Сравнение строк. Класс Vector. Конструкторы класса. Операторы. Функции члены. Доступ к вектору с помощью итератора. Вставка и удаление элементов из вектора. Сохранение в векторе объектов класса. Класс List. Шаблонная спецификация класса. Конструкторы класса. Операторы. Функции члены. Сортировка списка. Объединение списков. Хранение в списке объектов класса. Ассоциативные списки. Шаблонная спецификация класса. Конструкторы класса. Операторы. Функции члены. Распределитель памяти. Хранение объектов класса. Классы множество и мультимножество. Шаблонная спецификация. Конструкторы. Операторы. Функции члены. Классы стек и очередь. Шаблонная спецификация классов. Конструкторы классов. Операторы. Функции члены классов. Алгоритмы. Обзор алгоритмов библиотеки STL. Подсчет элементов. Удаление и замена. Реверсирование последовательности. Преобразование последовательности. Исследование алгоритмов.

**Описание практико-ориентированных заданий и кейсов**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Номер темы/модуля** | **Наименование практического занятия** | **Описание** |
| **1.**3 | Основные этапы разработки приложений. Способы описания алгоритмов | Составление алгоритмов | Составить алгоритм решения задачи и записать его различными способами (словесный, блок-схема, структурограмма, программа и т.д.) |
| **1.**4 | Тестирование и отладка | Использование интегрированных средств разработки для создания программных продуктов, тестирование и отладка программного кода | Составить программу для решения задачи с использованием интегрированных средств разработки, произвести ее отладку программными средствами и тестирование |
| 2.1 | Элементы языка. Структура программы. Операторы | Программирование алгоритмов линейной и разветвляющейся структуры | Составить программу для решения задачи на программирование с использованием условных операторов, переключателей, ввода и вывода, арифметических операций, стандартных математических функций средствами языков С/С++ |
| 2.1 | Элементы языка. Структура программы. Операторы | Программирование алгоритмов циклической структуры | Составить программу для решения задачи с использованием циклов средствами языков С/С++ |
| 2.2 | Массивы. Функции. Указатели. Структуры. Файлы | Обработка массивов | Составить программу для решения задачи на обработку данных, хранящихся в массивах средствами языков С/С++ |
| 2.2 | Массивы. Функции. Указатели. Структуры. Файлы | Работа с функциями | Составить программу для решения задачи с применением функций, определяемых пользователями |
| 2.2 | Массивы. Функции. Указатели. Структуры. Файлы | Определение структурных типов и обработка структур | Определить структурный тип и составить программу решения задачи на обработку структур |
| 2.2 | Массивы. Функции. Указатели. Структуры. Файлы | Работа с файлами | Составить программу для решения задачи на обработку данных, хранящихся в файлах с выводом результатов в файлы |
| 2.3 | Основы объектно-ориентированного программирования в С++ | Создание классов и работа с ними. Инкапсуляция | Определить классовый тип и составить программу решения задачи на обработку данных с использование объектов класса |
| 2.3 | Основы объектно-ориентированного программирования в С++ | Перегрузка операторов и функций. Полиморфизм | Определить классовый тип и составить программу решения задачи на обработку данных с использованием перегрузки операторов и функций относительно класса |
| 2.3 | Основы объектно-ориентированного программирования в С++ | Виртуальные функции. Динамический полиморфизм | Написать программу решения задачи с использованием наследования классов и виртуальных функций с применением динамического полиморфизма |
| 2.4 | Библиотека стандартных шаблонов | Использование при программировании библиотеки стандартных шаблонов | Написать программу решения задачи с использованием функций библиотек стандартных шаблонов Vector, List |

**8.Оценочные материалы по образовательной программе**

**8.1. Вопросы тестирования по модулям**

|  |  |  |  |
| --- | --- | --- | --- |
| **№ модуля** | **Вопросы входного тестирования** | **Вопросы промежуточного тестирования** | **Вопросы итогового тестирования** |
| 1 | Есть ли у вас опыт самостоятельной разработки программ на языке высокого уровня | Какие существуют способы описания алгоритмов? | По каким параметрам можно оценить оптимальность кода? |
| 2 | Какой язык программирования вы используете наиболее часто в своей профессиональной деятельности | Из каких основных алгоритмических структур могут быть составлены программы? | Почему при составлении алгоритмов и написании программного кода не приветствуется использование операторов перехода (goto)? |
| 3 | Какие принципы объектно-ориентированного программирования вы знаете? | Перечислите известные вам этапы разработки ПО | Какие стандартные библиотеки языков программирования С/С++ вам известны |
| 4 | По каким параметрам вы оцениваете оптимальность кода? | Назовите известные вам интегрированные средства разработки для создания программных продуктов на С++ | Что означает void в С++? |
| 5 | Часто ли вы применяете при разработке программ функции (подпрограммы)? | В чем отличия между компиляторами и интерпретаторами | Что такое рекурсия? |
| 6 | Какие интегрированные средства разработки вы применяете для создания программных продуктов? | Какие свойства алгоритма на ваш взгляд самые важные? | Можно ли в языках С/С++ создать программу без использования функций? |
| 7 | По каким причинам программа может выполняться бесконечно? | Какие категории ошибок может определить компилятор? | Для чего используются шаблоны функций? |
| 8 | Что подразумевает процесс тестирования программы? | Что означает эффективность алгоритма (программы) по времени? | Для чего в С++ применяются виртуальные функции |
| 9 | Для чего применяются логические операции в языках программирования? | Что означает эффективность алгоритма (программы) по памяти? | Какие предельные случаи нужно рассмотреть при тестировании программы, реализующей алгоритм нахождения корней уравнения ax+b=0? |
| 10 | Какие типы называются в языках программирования стандартными? | Для чего нужен дружественный интерфейс? | Какие препроцессорные директивы вам известны? |
| 11 | Как вы понимаете понятие "дружественный интерфейс" | Для чего нужен компилятор? | Что такое указатели? |
| 12 | Для каких целей в языках программирования введены несколько видов циклов | Насколько важным является свойство дискретности алгоритма и почему? | Какие логическая операции вам известны, запишите их обозначение в С++? |
| 13 | Какие функции называются рекурсивными? | Что означает фраза "зацикливание программы"? | Какие принципы объектно-ориентированного программирования вам известны? |
| 14 | Создавали ли вы собственные типы данных, какие? | Перечислите известные вам алгоритмы обработки массивов | Для чего используется деструктор? |
| 15 | Какие основные алгоритмические структуры вам известны | Какие языки высокого уровня вам известны? | Зачем нужно перегружать операторы? |
| 16 | Какие основные свойства алгоритма вам известны? | Какие языки низкого уровня вам известны? | Какие спецификаторы для разграничения доступа в классе вы знаете? |
| 17 | Перечислите известные вам этапы разработки ПО | Учитываются ли особенности компьютерных архитектур в языках программирования высокого уровня? | Может ли в классе быть несколько конструкторов, когда? |
| 18 | В чем суть методов восходящего и нисходящего программирования | При переносе программы на языке низкого уровня на другую аппаратную платформу есть ли необходимость ее переписывать | Можно ли в С++ создать собственные типы данных, каким образом? |
| 19 | Перечислите известные вам стандартные алгоритмы | В чем суть метода восходящего программирования | Какой класс называется абстрактным? |
| 20 | Учитываются ли особенности компьютерных архитектур в языках программирования высокого уровня? | В чем суть метода нисходящего программирования | Что работает быстрее компиляторы или интерпретаторы? |

**8.2.**  Показатели оценивания

Показателями оценивания являются правильность ответов на вопросы теста, работоспособность и эффективность разработанного слушателем программного кода.

При проведении входного тестирования зачисление слушателей осуществляется при наличии 40% правильных ответов (более 7 верных ответов из 20).

Шкала оценивания результатов промежуточной аттестации в форме тестирования:

|  |  |
| --- | --- |
| % правильных ответов | Результат оценки уровня компетенции |
| от 95 % | базовый уровень |
| 75%-94% | базовый уровень |
| 40%-74% | начальный уровень |
| менее 40% | не достигнут минимальный уровень |

Шкала оценивания результатов итоговой аттестации в форме тестирования и решения практического задания на написание программного кода:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| % правильных ответов | Правильность и эффективность программного кода | | | |
| Работоспособная программа, высоко эффективный алгоритм, дружественный интерфейс | Работоспособная программа, эффективный алгоритм | Работоспособная программа, неэффективный алгоритм | Не работоспособная программа |
| от 95 % | профессиональный уровень | базовый уровень | базовый уровень | возможно повторное выполнение практического задания/не аттестован |
| 75%-94% | продвинутый уровень | базовый уровень | базовый уровень | возможно повторное выполнение практического задания/не аттестован |
| 40%-74% | базовый уровень | начальный уровень | начальный уровень | возможно повторное выполнение практического задания/не аттестован |
| менее 40% | возможна повторная пересдача теста/не аттестован | возможна повторная пересдача теста/не аттестован | возможна повторная пересдача теста/не аттестован | не аттестован |

**.**

**8.3.**  **примеры контрольных заданий по модулям или всей образовательной программе**

Циклы

Дано натуральное число N. Вычислить:

Дано натуральное число N. Вычислить: .

Дано натуральное число N. Вычислить произведение первых N сомножителей .

Одномерные массивы

Дан массив натуральных чисел. Найти сумму элементов, кратных данному числу К.

В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.

Дана последовательность чисел a1, а2, ..., аn. Выяснить, какое число встречается раньше положительное или отрицательное.

Двумерные массивы

Задана матрица порядка n и число k. Разделить элементы k-й строки на диагональный элемент, расположенный в этой строке.

Для целочисленной квадратной матрицы найти число элементов, кратных k, и наибольший из полученных результатов.

Для целочисленной квадратной матрицы найти число элементов, кратных k, и их произведение.

Функции

Треугольник задан координатами своих вершин. Составить программу вычисления его площади.

Составить программу нахождения наибольшего общего делителя и наименьшего общего кратного двух натуральных чи­сел.

Написать программу нахождения суммы большего и меньше­го из трех чисел.

Обработка текстовой информации

Дана строка. Подсчитать в ней количество вхождений букв f, k,t.

Дана строка. Определить, сколько в ней символов \* ; :.

Дана строка, содержащая текст. Найти длину самого корот­кого слова и самого длинного слова.

Определение классов. Наследование. Виртуальные функции

Создать класс «Треугольник», заданный координатами вершин. Определить производные классы «Равносторонний» и «Прямоугольный» треугольники. Предусмотреть виртуальную функцию для вычисления площади описанной около треугольника окружности, сравнить площади окружностей, описанных около равностороннего и прямоугольного треугольников, координаты которых введены с клавиатуры.

Создать класс «Окружность», заданный координатами центра и точкой на окружности. Определить производный класс «Описанная окружность». Предусмотреть виртуальную функцию для вычисления периметра равностороннего треугольника, вписанного в окружность. Сравнить периметры 3 равносторонних треугольников, вписанных в окружности, данные о которых вводятся с клавиатуры.

Создать класс «Шар», заданный координатами центра и точкой на сфере. Определить производный класс «Окружность». Предусмотреть виртуальную функцию для вычисления площади круга, сравнить площади 2 кругов, данные о которых введены с клавиатуры.

Потоковый ввод-вывод. Текстовые и бинарные потоки. Перегрузка ввода-вывода

В одном файле хранятся сведения о книгах: код, автор, название, во 2 файле сведения об издании: код, издательство, год издания. Вывести на экран информацию о книгах автора, введенного с клавиатуры, изданных в 2013 году, введенного с клавиатуры (ФИО, должность, зарплата за год), выбрать сотрудников, получивших зарплату за год меньше средней по отделу.

В одном файле хранятся сведения о сотрудниках: ФИО, табельный номер, должность, отдел, во 2 файле сведения о работе за год: табельный номер, заработная плата по месяцам. Вывести на экран информацию о сотрудниках отдела, введенного с клавиатуры (ФИО, должность, зарплата за год), выбрать сотрудников, получивших зарплату за год меньше средней по отделу.

В одном файле хранятся сведения о товарах в магазине: код товара, название, цена, во 2 файле сведения о производстве: код товара, производитель, дата производства, срок годности. Вывести на экран информацию о товарах произведенных после 20.06.2012 (название, цена, производитель, дата производства), которые стоят менее 300 рублей, у которых истек срок годности

**.**

**8.4.**  **Тесты и обучающие задачи (кейсы), иные практикоориентированные формы заданий**

ТЕСТ 1

1. Что означает тип void в С++

А) тип определяемый пользователем

В) тип переменной

С) функция не возвращает значений

D) функция не выполняет никаких действий

E) функция не имеет параметров

2. Комментарии в С++ не могут начинаться символами

А) /\*

В) {

С) \n

E) “

3. Вывод данных на экран монитора осуществляется:

А) input

В) write

С) cin

D) printf

E) cout

4. Ввод данных осуществляется:

А) scanf

В) read

С) include

D) cin

E) cout

5. Какие типы переменных определены в С++

А) real

В) integer

С) int

D) double

E) float

6. Чему равно значение переменной в С++ при ее определении

А) 1

В) 0

С) неизвестно

D) TRUE

E) NULL

7. Какими символами обозначается операция присваивания

А) =

В) ==

С) :=

D) !=

E) +=

8. Каким символом обозначается логическая операция «И»

А) &

В) and

С) |

D) ||

E) &&

9. Как в С++ нельзя организовать ветвление

А) условным оператором

В) оператором цикла

С) оператором присваивания

D) условной трехместной операцией

E) переключателем

10. Какой спецификатор используется для вещественных чисел типа float

А) d

В) f

С) c

D) e

E) нет верных ответов

11. Для получения адреса используется операция

А) &

В) ^

С) \*

D) &&

E) нет верных ответов

12. Какое значение примет переменная s после выполнения следующих операторов

s=0;

for (i=1;i<5;i++) s=s+i;

А)10

В) 5

С) -5

D) -15

E) нет верных ответов

13. Что такое рекурсия

А) адрес объекта

В) тип объекта

С) самовызывающаяся функция

D) значение функции

E) библиотечная функция

14. Как в памяти размещены элементы массива

А) зависит от типа массива

В) в смежных участках памяти

С) произвольным образом

D) начиная с адреса 1 элемента подряд

E) нет правильных ответов

15. Для чего используются шаблоны функций

А) создания функций, обрабатывающих однотипные данные

В) автоматизации создания функций, обрабатывающих разнотипные данные

С) создания меню из однотипных функций

D) нет верного ответа

16. Какие высказывания верны для циклических структур

А) используются для организации ветвлений

В) используются при повторяющихся вычислениях

С) используются при выводе элементов массивов

D) бывают 3 видов

Е) бывают 4 видов

17. Для строк в языке С++ верны высказывания:

А) для строк введен отдельный встроенный тип

В) это массив из символов

С) строки можно присваивать

D) строки можно сравнивать целиком

E) строки можно сравнивать лишь поэлементно

18. Инициализация массива

А) объявление массива

В) присваивание начальных значений при определении

С) изменение значений элементов

D) присваивание значения одного объекта другому объекту

E) нет правильных ответов

19. Какое значение примет переменная s после выполнения следующих операторов

int s=10,a=2;

for (int i=1; i<4;i++)

if (i!=a) s=s\*a\*a;

else s=a;

s=s/a;

А) 0 В) 4 С) 8 D) 1 E) нет верных ответов

20. Какое значение примет переменная p после выполнения следующих операторов

int p=20,a=14;

p<a ? p=a/3:p=p%a;

А) 1 В) 6.6 С) 6 D) 4

ТЕСТ 2

1. Выберите все истинные высказывания из предложенных ниже про ООП в С++:

в языке реализованы три принципа ООП;

в языке не реализованы принципы ООП;

в языке реализованы два принципа ООП;

в языке используется наследование;

в языке не используется полиморфизм.

2. Для чего используется конструктор

для создания объектов;

для инициализации объектов;

для удаления объектов;

для получения значений объектов.

3. Для чего используется деструктор

для создания объектов;

для инициализации объектов;

для удаления объектов;

для получения значений объектов.

4. Может ли в объявлении класса не быть деструктора

да;

нет;

5. Может ли в объявлении класса не быть конструктора

да \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(когда);

нет \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(почему);

в исключительных случаях \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(когда).

6. Могут ли дружественные функции иметь доступ к защищенной части класса

да;

нет;

в исключительных случаях.

7. Для чего используется оператор расширения области видимости:

определения функций-членов класса;

прямого доступа к компонентам класса;

косвенного доступа к компонентам класса.

8. Выберите все истинные высказывания, из предложенных ниже

при наследовании базовый класс приобретает свойства производного;

при наследовании производный класс приобретает свойства базового;

при наследовании производный и базовый класс становятся идентичными

9. Могут ли у конструктора быть параметры?

да;

нет;

в исключительных случаях.

10. Может ли шаблон класса не иметь параметров в скобках < >

да;

нет;

в исключительных случаях \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(когда?)

11. Для чего используется конструктор копий

для инициализации;

для присваивания;

для удаления объекта;

для создания объекта;

12. Могут ли в объявлении класса использоваться встраиваемые функции

да, любые функции;

нет, никакие функции не могут быть встраиваемыми;

встраиваемым может быть только конструктор;

встраиваемым может быть только деструктор;

любые функции, кроме конструктора и деструктора;

только конструктор и деструктор.

13. Какое условие является обязательным для присвоения одного объекта другому

у объектов должны быть одинаковые типы;

у объектов должны быть одинаковые компоненты;

у объектов должны быть одинаковые конструктор и деструктор;

можно присваивать объекты любых типов.

14. Конструктор может иметь аргументы по умолчанию?

да\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (для чего);

нет \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(почему);

может, но не все аргументы.

15. Деструктор может иметь аргументы по умолчанию?

да;

нет;

может, но не все аргументы.

16. Оператор перегружается

относительно класса;

относительно любого типа данных;

вне зависимости от типов данных.

17. Если оператор-функция — член класса перегружает бинарный оператор, сколько параметров будет у функции

два;

не будет;

один.

18. Если оператор-функция —дружественная к классу перегружает унарный оператор, сколько параметров будет у функции

два;

один.

ни одного

19. Для чего используется функция преобразования?

Для преобразования стандартного типа в объект;

Для преобразования объекта к стандартному типу;

Для преобразования функций внутри класса

Для преобразования типов внутри класса

20. Как происходит наследование открытых, закрытых и защищенных переменных при разных спецификаторах доступа?

**.**

**8.5.**  Описание процедуры оценивания результатов обучения

После изучения материала каждого модуля проводится промежуточная аттестация знаний и умений в форме тестирования. По окончании обучения проводится итоговое тестирование. Тестовые задания (промежуточный и итоговый этапы) соответствуют перечню контрольных вопросов (таблица 8.1). Итоговая аттестация проводится в виде зачета, включающего итоговое тестирование и выполнение практического задания по написанию программного кода. Проверка практического задания выполняется путем анализа правильности и эффективности программного кода автоматизированной тестирующей системой и (или) преподавателем. Анализ эффективности программного кода производится с использованием автоматизированных тестирующих систем на наборе тестов, которые учитывают время работы программы (эффективность по времени) и ограничения на использование памяти (эффективность по памяти). При оценке эффективности внимание уделяется использованию алгоритмов для оптимизации программного кода и учет случаев, когда программа перестает работать (переполнение памяти, невозможность выполнения операций и т.д. ). В случае невыполнения одного из этапов (тестирование/разработка программного кода) итоговой аттестации слушатель после консультации с преподавателем имеет право на однократную пересдачу этапа.

**.**

**9.Организационно-педагогические условия реализации программы**

**9.1. Кадровое обеспечение программы**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№**  **п/п** | **Фамилия, имя, отчество (при наличии)** | **Место основной работы и должность, ученая степень и ученое звание (при наличии)** | **Ссылки на веб-страницы с портфолио (при наличии)** | **Фото в формате jpeg** | **Отметка о полученном согласии на обработку персональных данных** |
| **1** | Ильин Дмитрий Владимирович | ФГБОУ ВО "ЧГУ им. И.Н. Ульянова", заведующий кафедрой математического и аппаратного обеспечения информационных систем, к. ф.-м.н, доцент | http://vt.chuvsu.ru/kafedra-matematicheskogo-i-apparatnogo-obespecheni/61-nauchno-pedagogicheskiy-sostav.html |  | имеется |
| **2** | Ильина Лариса Алексеевна | ФГБОУ ВО "ЧГУ им. И.Н. Ульянова", старший преподаватель | http://vt.chuvsu.ru/kafedra-matematicheskogo-i-apparatnogo-obespecheni/61-nauchno-pedagogicheskiy-sostav.html |  | имеется |

**9.2.Учебно-методическое обеспечение и информационное сопровождение**

|  |  |
| --- | --- |
| **Учебно-методические материалы** | |
| Методы, формы и технологии | Методические разработки,  материалы курса, учебная литература |
| В процессе обучения используются автоматизированные тестирующие системы для проверки правильности разработанного слушателем программного кода при выполнении учебных задач | Ильин Д. В. Язык программирования C++: курс лекций / Ильин Д. В., Ильина Л. А., [отв. ред. И. Т. Артемьев] ; Чуваш. гос. ун-т им. И. Н. Ульянова - Чебоксары: Изд-во Чуваш. ун-та, 2015. - 124с. |
| В процессе обучения применяются дистанционные технологии на базе вебинарной платформы университета, системы управления курсами Moodle | Ильин, Д. В. Языки программирования: лабораторный практикум [для 1 курса факультета информатики и вычислительной техники] / Д. В. Ильин, Л. А. Ильина ; [отв. ред. И. Т. Артемьев] ; Чуваш. гос. ун-т им. И. Н. Ульянова. - Чебоксары: Изд-во Чуваш. ун-та, 2015. - 55с |
| В процессе обучения применяются объяснительно-иллюстративный метод; репродуктивный метод, проблемное изложение материала и частично-поисковый метод |  |
| В процесс обучения выстроен в формах: лекции, практик, самостоятельной работы и аттестации |  |

|  |  |
| --- | --- |
| **Информационное сопровождение** | |
| Электронные  образовательные ресурсы | Электронные  информационные ресурсы |
| [http://window.edu.ru](http://window.edu.ru/) – полнотекстовая библиотека учебных и учебно-методических материалов | <https://docs.microsoft.com/ru-ru/cpp/cpp/cpp-language-reference?view=vs-2019> – справочник по языку С++ на сайте Microsoft. |
| [https://informatics.msk.ru](https://informatics.msk.ru/) – сайт с задачами по программированию, теоретическим материалом для изучения и встроенной тестовой системой | <http://www.cplusplus.com/> ­– справочник по языку С++ на английском языке. |
| https://[acmp.ru](http://acmp.ru/) – «Школа программиста» с тестирующей системой и разделами для обучения языку программирования С++ | https://ru.cppreference.com/w/ – справочник по языку С++ на русском языке |

**9.3.Материально-технические условия реализации программы**

|  |  |
| --- | --- |
| Вид занятий | Наименование оборудования,  программного обеспечения |
| Лекция | ПЭВМ с доступом в Интернет, оборудованных видеокамерой и микрофоном |
| Практическое занятие, аттестация | ПЭВМ с доступом в Интернет, оборудованных видеокамерой и микрофоном и установленными средами программирования на языках С/С++:  Свободно распространяемая интегрированная среда для разработки приложений на языке С++ **Dev-C++ 5.0 beta 9.2 (4.9.9.2) with Mingw/GCC 3.4.2**  (<http://www.bloodshed.net/devcpp.html>)  Microsoft Visual Studio Community ([https://www.visualstudio.com/ru/vs/community](https://www.visualstudio.com/ru/vs/community/)) |
| Самостоятельная работа | ПЭВМ с доступом в интернет и установленным программным обеспечением:  **Dev-C++ 5.0 beta 9.2 (4.9.9.2) with Mingw/GCC 3.4.2**  Свободно распространяемая интегрированная среда для разработки приложений на языке С++(<http://www.bloodshed.net/devcpp.html>)  Microsoft Visual Studio Community ([https://www.visualstudio.com/ru/vs/community](https://www.visualstudio.com/ru/vs/community/) |

**III.Паспорт компетенций (Приложение 2)**

Описание перечня профессиональных компетенций, качественное изменение которых осуществляется в результате обучения.

Планируемые результаты обучения должны быть определены в виде знаний, умений, навыков и (или) опыта деятельности, характеризующих формирование/развитие компетенции(-й) в области цифровой экономики и представлены в виде Паспорта компетенций в машиночитаемом текстовом формате. Структура паспорта представлена в приложении.

ПАСПОРТ КОМПЕТЕНЦИИ

(Программирование на С/С++)

(ФГБОУ ВО "Чувашский государственный университет имени И.Н. Ульянова")

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Наименование компетенции | | способен разрабатывать и осуществлять отладку программного кода | |
| 2. | Указание типа компетенции | общекультурная/  универсальная |  | |
| общепрофессиональ ная |  | |
| профессиональная | профессиональная | |
| профессионально-специализированная |  | |
| 3. | Определение, содержание и основные сущностные характеристики компетенции | | осуществляет формализацию и алгоритмизацию поставленных задач; разрабатывает программный код с использованием языков программирования; проверяет и отлаживает программный код | |
| 4. | Дескриптор знаний, умений и навыков по уровням | | Уровни сформирован ности компетенции  обучающегося | Индикаторы |
| Знать: Стандартные методы и приемы алгоритмизации; Синтаксис языка программирования, типовые структуры, стандартные библиотеки; Типы и форматы сообщений об ошибках; Компиляторы и отладчики программного кода  Уметь: Использовать стандартные методы и приемы алгоритмизации; Применять стандартные алгоритмы; Разрабатывать программный код с использованием языков программирования С/С++; Применять методы и приемы отладки программного кода.  Владеть навыками: Формализации и алгоритмизации поставленных задач; Разработки программ; Отладки программного кода | | Начальный уровень  (Компетенция недостаточно развита. Частично проявляет навыки, входящие в состав компетенции. Пытается, стремится проявлять нужные навыки, понимает их необходимость, но у него не всегда получается.) | Способен перечислить некоторые методы и приемы алгоритмизации, формы записи алгоритмов.  Знает синтаксис языка программирования С/С++, стандартные библиотеки. Применяет простые стандартные приемы алгоритмизации.  Составляет простые программы. Использует компилятор и отладчик для проверки программ и поиска ошибок, не всегда может их правильно интерпретировать. Разработанные программы не эффективны. |
| Знать: Методы и приемы алгоритмизации поставленных задач; Алгоритмы решения типовых задач, области и способы их применения; Синтаксис языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; Типы и форматы сообщений об ошибках, предупреждений. Современные компиляторы, отладчики и оптимизаторы программного кода  Уметь: Использовать методы и приемы алгоритмизации поставленных задач; Применять стандартные алгоритмы в соответствующих областях; Разрабатывать программный код с использованием языков программирования С/С++; Применять методы и приемы отладки программного кода  Владеть навыками: Формализации и алгоритмизации поставленных задач; Разработки эффективных программ; Применения методов и приемов отладки программного кода | | Базовый уровень  (Уверенно владеет навыками, способен, проявлять соответствующие навыки в ситуациях с элементами неопределён ности, сложности.) | Применяет различные методы и приемы алгоритмизации, разные формы записи алгоритмов. Использует синтаксические элементы и конструкции языка программирования, стандартные библиотеки. Составляет работоспособные программы. Использует современные компиляторы и отладчики для проверки программ и поиска ошибок, может их правильно интерпретировать. Разработанные алгоритмы и программы достаточно эффективны. |
| Знать: Технологии и методы алгоритмизации и программирования; Синтаксис и семантику языков программирования высокого уровня С/С++, объектно-ориентированный подход; Типы и форматы сообщений об ошибках, предупреждений. Современные компиляторы, отладчики и оптимизаторы программного кода  Уметь: Использовать методы и приемы алгоритмизации поставленных задач; Применять эффективные алгоритмы в соответствующих областях; Разрабатывать оптимальный и эффективный программный код с использованием средств языков программирования С/С++; Применять методы и приемы отладки и оптимизации программного кода;  Владеть навыками: Формализации и алгоритмизации поставленных задач; Разработки эффективных программ; Применения методов и приемов отладки и оптимизации программного кода. | | Продвинутый  (Владеет сложными навыками, способен активно влиять на происходящее, проявлять соответствующие навыки в ситуациях повышенной сложности.) | Применяет различные технологии и методы алгоритмизации программирования. Создает собственные структуры данных и типы данных средствами языка программирования. Использует библиотеки стандартных шаблонов. Составляет сложные программы с дружественным интерфейсом. Использует современные компиляторы и отладчики для проверки и оптимизации программ, поиска ошибок.  Осуществляет отладку программного кода на уровне  программных модулей.  Разработанные алгоритмы и программы эффективны и по времени и по памяти. |
| Знать: Технологии и методы алгоритмизации и программирования; Син таксис и семантику языков программирования высокого уровня С/С++, объектно-ориентированный подход; Языки формализации функциональных спецификаций; Типы и форматы сообщений об ошибках, предупреждений; Современные компиляторы, отладчики и оптимизаторы программного кода; Компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними  Уметь: Использовать методы и приемы алгоритмизации поставленных задач с использованием специальных языков формализации функциональных спецификаций; Применять эффективные алгоритмы в соответствующих областях; Разрабатывать оптимальный и эффективный программный код с использованием средств языков программирования С/С++; Применять методы и приемы отладки и оптимизации программного кода; Осуществлять отладку программного кода на уровне программных модулей, а также межмодульных взаимодействий и взаимодействий с окружением. Использовать возможности имеющейся технической и/или программной архитектуры  Владеть навыками: Использования технологий и методов алгоритмизации и программирования при решении поставленных задач с использованием специальных языков формализации функциональных спецификаций; Разработки эффективных алгоритмов и программ в соответствующих областях; Осуществления отладку программного кода на уровне программных модулей, а также межмодульных взаимодействий и взаимодействий с окружением. Использовать возможности имеющейся технической и/или программной архитектуры | | Профессиональный  (Владеет сложными навыками, создает новые решения для сложных проблем со многими взаимодействую-щими факторами, предлагает новые идеи и процессы, способен активно влиять на происходящее, проявлять соответствующие навыки  в ситуациях повышенной сложности.) | Знает методы и алгоритмы разработки проектов, реализуемых группами разработчиков  Осуществляет отладку программного кода на уровне  программных модулей, межмодульных взаимодействий и взаимодействий с окружением.  Использует возможности имеющейся технической и/или программной архитектуры.  Разработанные алгоритмы и программы имеют высокую эффективность и дружественный интерфейс |
| 5. | Характеристика взаимосвязи данной компетенции с другими компетенциями/ необходимость владения другими компетенциями для формирования данной компетенции | | Для формирования данной компетенции необходимо владение общепрофессиональной компетенцией:  способность использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности | |
| 6. | Средства и технологии оценки | | Для оценки сформированности компетенции используются входной, промежуточный и выходной контроль знаний: тесты, практические задания | |

**VI.Иная информация о качестве и востребованности образовательной программы** (результаты профессионально-общественной аккредитации образовательной программы, включение в системы рейтингования, призовые места по результатам проведения конкурсов образовательных программ и др.) (при наличии)

Имеется профессионально-общественная аккредитация образовательной программы0РР000000181 от 26 июня 2018 года Ассоциации "Русский Регистр"по направлению подготовки 09.03.01 Информатика и вычислительная техника, в которой представлена дисциплина "Программирование" аналогичная по содержанию

**V.Рекомендаций к программе от работодателей**: наличие не менее двух писем и/или подтверждения на цифровой платформе Государственной системы предоставления ПЦС от работодателей о рекомендации образовательной программы для реализации в рамках Государственной системы предоставления ПЦС на формирование у трудоспособного населения компетенций цифровой экономики с указанием востребованности результатов освоения программы в сфере деятельности соответствующих компаний и готовности к рассмотрению заявок наиболее успешно освоивших образовательную программу граждан на прохождение стажировки и (или) собеседования на предмет трудоустройства путем проставления отметки в профиле программы

Имеются письма от **ООО "**Кейсистемс**"** г. Чебоксары и ООО "КСБ-СОФТ" г. Чебоксары о рекомендации образовательной программы для реализации в рамках Государственной системы предоставления ПЦС на формирование у трудоспособного населения компетенций цифровой экономики с указанием востребованности результатов освоения программы в сфере деятельности соответствующих компаний.

**VI.Указание на возможные сценарии профессиональной траектории граждан** по итогам освоения образовательной программы (в соответствии с приложением)

Развитие компетенций в текущей сфере занятости

**VII.Дополнительная информация**

**VIII.Приложенные Скан-копии**

Утвержденной рабочей программа (подпись, печать, в формате pdf)