

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ЭКОНОМИКИ И СЕРВИСА

**РАБОЧАЯ ПРОГРАММА
УЧЕБНОЙ ДИСЦИПЛИНЫ**

ОП.05 Основы программирования

программы подготовки специалистов среднего звена
09.02.03 Программирование в компьютерных системах

на базе среднего общего образования

Форма обучения: *очная*

Находка 2020

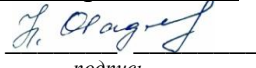
Рабочая программа учебной дисциплины *ОП.05 «Основы программирования»* разработана в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.03 Программирование в компьютерных системах, утвержденного приказом Минобрнауки России от «28» июля 2014 г. № 804, примерной образовательной программой.

Разработчик(и):

Коломийцев А.К., преподаватель ОСПО филиала ФГБОУ ВО «ВГУЭС» в г. Находке

Рассмотрено и одобрено на заседании межпредметной цикловой методической комиссии

Протокол № 9 от «30» марта 2020 г.

Председатель ЦМК  Н.П. Фадеева
подпись

СОДЕРЖАНИЕ

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

1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1 Место дисциплины в структуре основной образовательной программы

Учебная дисциплина *ОП.05 «Основы программирования»* является частью профессионального учебного цикла основной образовательной программы (далее ООП) в соответствии с ФГОС СПО по специальности *09.02.03 Программирование в компьютерных системах*.

1.2 Цель и планируемые результаты освоения дисциплины

По итогам освоения дисциплины, обучающиеся должны продемонстрировать результаты обучения, соотнесённые с результатами освоения ООП СПО, приведенные в таблице.

Код компетенции	Наименование общих компетенций	Результаты освоения
ОК 1	Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	практический опыт: - выполнять профессиональные задачи при выполнении выпускной квалификационной работы; - проявлять творческую инициативу, демонстрировать профессиональную подготовку умения: - овладеть первичными профессиональными навыками и умениями; - планировать будущую профессиональную деятельность знания: - иметь представление о будущей профессии; - ориентироваться в маршруте студента по специальности; - называть основные виды работ, выполняемые при работе по специальности
ОК 2	Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	практический опыт: - планирования деятельности, применяя технологию с учетом изменения параметров объекта; - выбирать типовой способ (технологию) решения задачи в соответствии с заданными условиями, имеющимися ресурсами, критериями качества и эффективности умения: - планировать деятельность по решению задачи в рамках заданных (известных) технологий, в том числе выделяя отдельные составляющие технологии; - анализировать потребности в ресурсах и планировать ресурсы в соответствии с

		<p>заданным способом решения задачи</p> <p>знания:</p> <ul style="list-style-type: none"> - методы и способы выполнения профессиональных задач; - называть ресурсы для решения поставленной задачи в соответствии с заданным способом деятельности
ОК 3	<p>Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.</p>	<p>практический опыт:</p> <ul style="list-style-type: none"> - проводить анализ причин существования проблемы; - предлагать способ коррекции деятельности на основе результатов текущего контроля и результатов оценки продукта деятельности; - определять показатели результативности деятельности в соответствии с поставленной профессиональной задачей; - задавать критерии для определения способа разрешения проблемы; - прогнозировать последствия принятых решений; - называть риски на основе самостоятельно проведенного анализа ситуации; - предлагать способы предотвращения и нейтрализации рисков <p>умения:</p> <ul style="list-style-type: none"> - самостоятельно задавать критерии для анализа рабочей ситуации на основе эталонной ситуации и определять проблему; - планировать текущий контроль своей деятельности в соответствии с заданной технологией деятельности и определенным результатом (целью) или продуктом деятельности; - определять проблему на основе самостоятельно проведенного анализа ситуации; - планировать и оценивать продукт своей деятельности на основе заданных критериев; - определять критерии оценки продукта на основе задачи деятельности; - выбирать способ разрешения проблемы в соответствии с заданными критериями и ставить цель деятельности; - оценивать последствия принятых решений; - анализировать риски (определять степень вероятности и степень влияния на достижение цели) и

		<p>обосновывать достижимость цели</p> <p>знания:</p> <ul style="list-style-type: none"> - технологии анализа рабочей ситуации в соответствии с заданными критериями, указывая ее соответствие/несоответствие эталонной ситуации; - принципы осуществления текущего контроля своей деятельности по заданному алгоритму; - способы оценивания продукта своей деятельности по характеристикам
ОК 4	<p>Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личного развития.</p>	<p>практический опыт:</p> <ul style="list-style-type: none"> - предлагать источник информации определенного типа/конкретный источник для получения недостающей информации и обосновывать свое предложение; - характеризовать произвольно заданный источник информации в соответствии с задачей деятельности; - принимает решение о завершении/продолжении информационного поиска на основе оценки достоверности/непротиворечивости полученной информации; - делать вывод о применимости общей закономерности в конкретных условиях; - делать вывод на основе предоставленных эмпирических или статистических данных <p>умения:</p> <ul style="list-style-type: none"> - самостоятельно находить источник информации по заданному вопросу, пользуясь электронным или бумажным каталогом, электронным или бумажным каталогом, справочно-библиографическими пособиями, поисковыми системами Интернета; - указывать недостаток информации, необходимой для решения задачи; - формулировать вопросы, нацеленные на получение недостающей информации; - извлекать информацию по двум и более основаниям из одного или нескольких источников и систематизировать ее в рамках заданной структуры; - делать выводы об объектах, процессах, явлениях на основе сравнительного анализа информации и них по заданным критериям; - задавать критерии для сравнительного анализа информации в соответствии с

		<p>поставленной задачей деятельности</p> <p>знания:</p> <ul style="list-style-type: none"> - выделять из содержащего избыточную информацию источника информацию, необходимую для решения задачи; - выделять в источнике информации вывод и/или аргументы
ОК 5	Использовать информационно-коммуникационные технологии в профессиональной деятельности.	<p>практический опыт:</p> <ul style="list-style-type: none"> - применять ИКТ при выполнении профессиональных задач <p>умения:</p> <ul style="list-style-type: none"> - применять ИКТ при выполнении заданий <p>знания:</p> <ul style="list-style-type: none"> - перечислять ИКТ, применяемые в профессиональной деятельности; - ориентироваться в информационно-коммуникационных технологиях, применяемых в профессиональной деятельности
ОК 6	Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.	<p>практический опыт:</p> <ul style="list-style-type: none"> - принимать и фиксировать решение по вопросам для группового обсуждения; - фиксировать особые мнения; - использовать приемы выхода из ситуации, когда дискуссия зашла в тупик; - давать сравнительную оценку идей, высказанных участниками группы, относительно цели групповой работы; - самостоятельно готовить средства наглядности; - самостоятельно выбирать жанр монологического высказывания в зависимости от его цели и целевой аудитории; - запрашивать мнение партнера по диалогу; - извлекать из устной речи (монолог, диалог, дискуссия) фактическую и оценочную информацию, определять основную тему, предложения, аргументы, доказательства, выводы, оценки; - самостоятельно определять жанр письменной коммуникации в зависимости от цели; - создавать продукт письменной коммуникации сложной конструкции <p>умения:</p> <ul style="list-style-type: none"> - договариваться о процедуре и вопросах для обсуждения в группе в соответствии с поставленной целью деятельности команды (группы);

		<ul style="list-style-type: none"> - при групповом обсуждении задавать вопросы, проверять адекватность понимания идей других; - соблюдать заданный жанр высказывания (служебный доклад, выступление на совещании, собрании, презентации товара (услуги)); - использовать средства наглядности или невербальные средства, направленные на выяснение мнения (позиции); - задавать вопросы, направленные на выяснение фактической информации; - создавать стандартный продукт письменной коммуникации <p>знания:</p> <ul style="list-style-type: none"> - правила участия в групповом обсуждении, высказываясь в соответствии с заданной процедурой и по заданному вопросу; - соблюдать нормы публичной речи и регламент, используя паузы для выделения смысловых блоков своей речи; - начинать и заканчивать служебный разговор в соответствии с нормами; - отвечать на вопросы, направленные на выяснение фактической информации; - извлекать из устной речи (монолог, диалог, дискуссия) основное содержание фактической информации
ОК 7	Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.	<p>практический опыт:</p> <ul style="list-style-type: none"> - контролировать и отвечать за работу занимающихся <p>умения:</p> <ul style="list-style-type: none"> - анализировать работу занимающихся и результат выполненного задания; - оценивать работу и контролировать работу занимающихся <p>знания:</p> <ul style="list-style-type: none"> - выполнять поставленные задания, являясь членом группы
ОК 8	Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.	<p>практический опыт:</p> <ul style="list-style-type: none"> - анализировать собственные мотивы и внешнюю ситуацию при принятии решений, касающихся своего продвижения <p>умения:</p> <ul style="list-style-type: none"> - указывает «точки успеха» и «точки роста»; - указывает причины успехов и неудач в деятельности; - анализировать/формулировать запрос на внутренние ресурсы (знания, умения,

		<p>навыки, способы деятельности, ценности, установки) для решения профессиональной задачи</p> <p>знания:</p> <ul style="list-style-type: none"> - называть трудности, с которыми столкнулся при решении задачи и знать пути их преодоления/избегания в дальнейшей деятельности
ОК 9	Ориентироваться в условиях частой смены технологий в профессиональной деятельности.	<p>практический опыт:</p> <ul style="list-style-type: none"> - применять современные технологии в профессиональной деятельности <p>умения:</p> <ul style="list-style-type: none"> - сравнивать технологии, применяемые в профессиональной деятельности; - выбирать технологии для своей профессиональной деятельности <p>знания:</p> <ul style="list-style-type: none"> - информацию о современных технологиях в профессиональной деятельности
ПК 1.1	Выполнять разработку спецификаций отдельных компонент.	<p>иметь практический опыт:</p> <ul style="list-style-type: none"> - разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования; - разработки кода программного продукта на основе готовой спецификации на уровне модуля; - использования инструментальных средств на этапе отладки программного продукта; - проведения тестирования программного модуля по определенному сценарию; <p>уметь:</p> <ul style="list-style-type: none"> - осуществлять разработку кода программного модуля на современных языках программирования; - создавать программу по разработанному алгоритму как отдельный модуль; - выполнять отладку и тестирование программы на уровне модуля; - оформлять документацию на программные средства; - использовать инструментальные средства для автоматизации оформления документации; <p>знать:</p> <ul style="list-style-type: none"> - основные этапы разработки программного обеспечения; - основные принципы технологии структурного и объектно-
ПК 1.2	Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.	
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств.	
ПК 1.4	Выполнять тестирование программных модулей.	
ПК 1.5	Осуществлять оптимизацию программного кода модуля.	

		<p>ориентированного программирования; – основные принципы отладки и тестирования программных продуктов; – методы и средства разработки технической документации</p>
ПК 3.1	<p>Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.</p>	<p>иметь практический опыт:</p> <ul style="list-style-type: none"> – участия в выработке требований к программному обеспечению; – участия в проектировании программного обеспечения с использованием специализированных программных пакетов; <p>уметь:</p> <ul style="list-style-type: none"> - владеть основными методологиями процессов разработки программного обеспечения; – принимать участие в формировании требований к ПО, выполнять анализ и спецификацию требований, уточнение требований на работающих прототипах; – выполнять системный анализ и проектирование компонент ПО на основе существующих методологий с использованием автоматизированных программных (CASE) средств; – выполнять интеграцию отдельных компонент ПО в единую программную систему, осуществлять их коммуникацию и взаимодействие друг с другом, а также другими программными средствами; – участвовать в разработке и формализованном описании тестовых сценариев, выполнять тестирование ПО на основе разработанных спецификаций; – выполнять отладку компонент ПО с использованием специализированных программных средств; – принимать участие в предпродажной подготовке, внедрении и сопровождении ПО; – выполнять формализованное описание компонент ПО, формировать техническую и эксплуатационную документацию на основе принятых стандартов с использованием специализированных программных пакетов – использовать методы для получения кода с заданной функциональностью и степенью качества.;

		<p>– выполнять кодирование компонент ПО на основе разработанных спецификаций и существующих стандартов с использованием современных инструментальных средств разработки (ИСР);</p> <p>- производить инспектирование качества и эффективности программного кода, степени его соответствия стандартам кодирования, выполнять оптимизацию и ревьюирование программного кода с использованием специализированных программных пакетов.</p> <p>знать:</p> <p>– модели процесса разработки программного обеспечения;</p> <p>– основные принципы процесса разработки программного обеспечения;</p> <p>– основные подходы к интегрированию программных модулей;</p> <p>– основные методы и средства эффективной разработки;</p> <p>– основы верификации и аттестации программного обеспечения;</p> <p>– концепции и реализации программных процессов;</p> <p>– принципы построения, структуры и приёмы работы с инструментальными средствами, поддерживающими создание программного обеспечения;</p> <p>– основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов;</p> <p>– стандарты качества программного обеспечения;</p> <p>– методы и средства разработки программной документации.</p>
--	--	--

2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1 Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Объем образовательной программы учебной дисциплины	260
в том числе:	
– теоретическое обучение	86

– практические занятия <i>(если предусмотрено)</i>	113
– лабораторные занятия <i>(если предусмотрено)</i>	
– курсовая работа (проект) <i>(если предусмотрено)</i>	
– самостоятельная работа	48
– консультации	13
– промежуточная аттестация – <i>(форма промежуточной аттестации)</i>	Другая форма контроля/Зачет/Экзамен

2.2 Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, формированию которых способствует элемент программы												
1	2	3	4												
Тема 1.	Основные принципы программирования	19	ОК 1												
	<table border="1"> <tr> <td data-bbox="387 587 439 628">1</td> <td data-bbox="439 587 1720 628">Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования.</td> </tr> <tr> <td data-bbox="387 628 439 670">2</td> <td data-bbox="439 628 1720 670">Исходный, объектный и загрузочный модули. Интегрированная среда программирования.</td> </tr> <tr> <td data-bbox="387 670 439 711">3</td> <td data-bbox="439 670 1720 711">Методы программирования: структурный, модульный, объектно-ориентированный.</td> </tr> <tr> <td data-bbox="387 711 439 753">4</td> <td data-bbox="439 711 1720 753">Достоинства и недостатки методов программирования.</td> </tr> </table>		1	Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования.	2	Исходный, объектный и загрузочный модули. Интегрированная среда программирования.	3	Методы программирования: структурный, модульный, объектно-ориентированный.	4	Достоинства и недостатки методов программирования.	ОК 2				
	1		Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования.												
	2		Исходный, объектный и загрузочный модули. Интегрированная среда программирования.												
3	Методы программирования: структурный, модульный, объектно-ориентированный.														
4	Достоинства и недостатки методов программирования.														
Практические занятия	ОК 3														
Самостоятельная работа обучающихся	ОК 4														
Тема 2.	Особенности выполнения программ	15	ОК 5												
	<table border="1"> <tr> <td data-bbox="387 1144 439 1185">1</td> <td data-bbox="439 1144 1720 1220">История развития языка программирования. Структурная схема программы на алгоритмическом языке. Лексика языка. Переменные и константы.</td> </tr> <tr> <td data-bbox="387 1220 439 1262">2</td> <td data-bbox="439 1220 1720 1262">Типы данных. Выражения и операции</td> </tr> <tr> <td data-bbox="387 1262 439 1303">3</td> <td data-bbox="439 1262 1720 1303">Операторы языка</td> </tr> <tr> <td data-bbox="387 1303 439 1345">4</td> <td data-bbox="439 1303 1720 1345">Массивы</td> </tr> <tr> <td data-bbox="387 1345 439 1386">5</td> <td data-bbox="439 1345 1720 1386">Строки и множества</td> </tr> <tr> <td data-bbox="387 1386 439 1428">6</td> <td data-bbox="439 1386 1720 1428">Процедуры и функции</td> </tr> </table>		1	История развития языка программирования. Структурная схема программы на алгоритмическом языке. Лексика языка. Переменные и константы.	2	Типы данных. Выражения и операции	3	Операторы языка	4	Массивы	5	Строки и множества	6	Процедуры и функции	ОК 6
	1		История развития языка программирования. Структурная схема программы на алгоритмическом языке. Лексика языка. Переменные и константы.												
	2		Типы данных. Выражения и операции												
	3		Операторы языка												
	4		Массивы												
	5		Строки и множества												
6	Процедуры и функции														
	ОК 7														
	ОК 8														
	ПК 1.1														
	ПК 1.2														
	ПК 1.3														
	ПК 1.4														
	ПК 1.5														
	ПК 3.1														

	Самостоятельная работа обучающихся	24	ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 3.1
Тема 3.	Формат и структура программы	16	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
	1 Формат и структура программы		
	2 Переменные, константы, типы данных		
	3 Операции и операторы		
	4 Процедуры ввода-вывода, условные операторы	20	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 3.1
	5 Циклы, процедуры и функции		
	6 Работа с файлами и указателями		
	7 Модуль Crt		
	8 Модуль Graph	11	
	Практические занятия		
	Задача на нахождение корней квадратного уравнения		
	Задача на зеркальное отображение символов строки относительно ее центра	15	ОК 1 ОК 2
	Задача на сортировку чисел в двумерном массиве (пузырьковая сортировка)		
	Чтение и запись данных из/в файл.		
	Программирование звукового генератора, разноцветный текст.	11	
	Построение графических объектов		
	Самостоятельная работа обучающихся	11	
Тема 4.	Основные принципы объектно-ориентированного программирования	15	ОК 1 ОК 2

	1	Основные принципы объектно-ориентированного программирования		ОК 3
	2	Формат и структура программы		ОК 4
	3	Переменные, константы, типы данных		ОК 5
	4	Операции и операторы		ОК 6
	5	Циклы, процедуры и функции		ОК 7
	6	Классы (поля, свойства, методы, события)		ОК 8
	7	Стандартные базовые классы.		ОК 9
	8	Визуальные компоненты.		ПК 1.1
	9	Не визуальные компоненты		ПК 1.2
	Практические занятия		40	ПК 1.3
	Разработать приложение «Калькулятор»			ПК 1.4
	Разработать приложение «Графический редактор»			ПК 1.5
	Самостоятельная работа обучающихся		10	ПК 3.1
Консультации			13	
Промежуточная аттестация (<i>другая форма контроля/зачет/дифференцированный зачет</i>)				
Всего:			260	

3 УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1 Материально-техническое обеспечение

Для реализации программы учебной дисциплины предусмотрено наличие следующих специальных помещений:

Лаборатория системного и прикладного программирования: количество посадочных мест – 30 шт., доска меловая – 1 шт., стол для преподавателя – 1 шт., стул для преподавателя – 1 шт., стол для преподавателя компьютерный – 1 шт.; количество персональных компьютеров с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации - 15 (Монитор 17" Acer, системный блок: процессор Intel(R) Pentium(R) Dual CPU E2180 @ 2.00GHz , ОЗУ 2 Гб, HDD 250 Гб), 1 экран Projecta, 1 проектор Sanyo PLC
ПО: Microsoft Windows 7 Professional Russian (Academic Open license 47882164), Microsoft Office 2007 RUS (бессрочно 44216302), Winrar (Лицензия RUK-web-1355405), Adobe Google Chrome (свободное); Adobe Acrobat Reader (свободное); Adobe Flash Player (свободное), Kaspersky Endpoint Security 10 для Windows (Номер лицензии 17E0200430130957417676), Java(TM) 6 Update 26 (свободное), справочно-правовая система КонсультантПлюс (Договор № 2020-А 0130)

3.2 Информационное обеспечение реализации программы

Для реализации программы учебной дисциплины библиотечный фонд ВГУЭС укомплектован печатными и электронными изданиями.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Основная литература

1. Баженова, И.Ю. Введение в программирование : курс лекций / Баженова И.Ю., Сухомлин В.А. — Москва : Интуит НОУ, 2016. — 411 с. <https://book.ru/book/917645>
2. Основы программирования : курс лекций / Мейер Бертран — Москва : Интуит НОУ, 2016. — 324 с. <https://book.ru/book/917903>
3. Алексеев, Е.Р. Программирование на языке C++ в среде Qt Creator : курс лекций / Алексеев Е.Р., Злобин Г.Г., Костюк Д.А., Чеснокова О.В., Чмыхало А.С. — Москва : Интуит НОУ, 2016. — 715 с. <https://book.ru/book/918128>

Дополнительная литература

1. Попов, А.А. Эргономика пользовательских интерфейсов в информационных системах : учебное пособие / Попов А.А. — Москва : КноРус, 2020. — 304 с. <https://book.ru/book/935936>
2. Макарова, Н.В. Основы программирования : учебник / Макарова Н.В., Нилова Ю.Н., Зеленина С.Б., Лебедева Е.В. — Москва : КноРус, 2020. — 451 с. <https://book.ru/book/936582>
3. Златопольский, Д.М. Программирование: типовые задачи, алгоритмы, методы : учебное пособие / Златопольский Д.М. 4-е изд. — Москва : Лаборатория знаний, 2020. — 224 с. — <https://book.ru/book/936428>

Электронные ресурсы

1. Научная электронная библиотека eLIBRARY.RU: <http://www.eLIBRARY.RU>
2. Ресурс Цифровые учебные материалы <http://abc.vvsu.ru/>
3. ЭБС «Руконт»: <http://www.rucont.ru/>
4. ЭБС «Юрайт»: <http://www.biblio-online.ru/>

4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
Умения:	
- работать в среде программирования;	практические занятия, индивидуальные задания, внеаудиторная самостоятельная работа
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	
Знания:	
- этапы решения задачи на компьютере;	Оценка выполнения устных и письменных групповых и индивидуальных заданий (доклад, реферат, презентация, эссе) Нетрадиционные формы контроля: - кроссворд; - головоломка; - ребус; - шарада; - викторина; Методы контроля: - метод тестирования; - проектный метод; - «мозговой штурм»; - «снежный ком»; - «аквариум».
- типы данных;	
- базовые конструкции изучаемых языков программирования;	
- принципы структурного и модульного программирования;	
- принципы объектно-ориентированного программирования.	

Для оценки достижения запланированных результатов обучения по дисциплине разработаны контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации, которые прилагаются к рабочей программе дисциплины.

МИНОБРНАУКИ РОССИИ

МИНОБРНАУКИ

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ЭКОНОМИКИ И СЕРВИСА

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА

для проведения текущего контроля и промежуточной аттестации
по учебной дисциплине

ОП.05 Основы программирования

программы подготовки специалистов среднего звена
09.02.03 Программирование в компьютерных системах

Форма обучения: очная

Находка 2020

Контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации по учебной дисциплине *ОП.05 «Основы программирования»* разработаны в соответствии с требованиями ФГОС СПО по специальности 09.02.03 Программирование в компьютерных системах, утвержденного приказом Минобрнауки России от «28» июля 2014 г. № 804, примерной образовательной программой.

Разработчик(и): Коломийцев А.К., преподаватель ОСПО филиала ФГБОУ ВО «ВГУЭС» в г. Находке.

Рассмотрена на заседании МПЦК от 30 марта 2020 г., протокол № 9

Председатель МПЦК  Фадеева Н.П.

1. Общие положения

Контрольно-оценочные средства (КОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу подготовки специалистов среднего звена ОП.05 Основы программирования.

КОС включают контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме контрольной работы.

КОС разработаны в соответствии с:

- программой подготовки специалистов среднего звена (далее - ППССЗ) по специальности 09.02.03 Программирование в компьютерных системах, реализуемой в колледже;
- программой подготовки специалистов среднего звена ОП.05 Основы программирования.

2. Результаты освоения дисциплины, подлежащие проверке

Результаты обучения (освоенные умения, усвоенные знания)	Основные показатели оценки результатов
ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	- демонстрация интереса к будущей профессии.
ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	- выбор и применение методов и способов решения профессиональных задач в области разработки и администрирования баз данных; - оценка эффективности и качества выполнения.
ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	- решение стандартных и нестандартных профессиональных задач в области разработки и администрирования баз данных.
ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.	- эффективный поиск необходимой информации; - использование различных источников, включая электронные.
ОК 5. Использовать информационнокоммуникационные технологии в профессиональной деятельности.	- разрабатывать, программировать и администрировать базы данных
ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.	- взаимодействие с обучающимися, преподавателями и мастерами в ходе обучения
ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.	- самоанализ и коррекция результатов собственной работы
ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.	- организация самостоятельных занятий при изучении профессионального модуля
ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.	- анализ инноваций в области разработки и администрирования баз данных
ПК 1.1. Выполнять разработку спецификаций отдельных компонент.	- создание отдельных компонент - выполнение спецификаций компонент
ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых	- выполнение создания кода программного продукта на уровне модуля в соответствии с требованиями к готовому программному продукту
ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.	- использование специализированных программных средств отладки программных модулей - разработка компонент программных модулей с использованием современных инструментальных средств и технологий.
ПК 1.4. Выполнять тестирование программных модулей.	- выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев. - определение ошибок в программном коде с использованием тестовых наборов
ПК 1.5. Осуществлять оптимизацию программного кода модуля.	- выявление избыточности кода программного продукта и его оптимизация.

	- анализ оптимизации программного кода модуля
ПК 3.1. Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.	- анализ проектной и технической документацию на уровне взаимодействия компонент программного обеспечения

3. Структура контрольных заданий

3.1. Задания текущего контроля

3.1.1 Практические задания

Тема 1.4. Среда программирования

Практическое задание № 1.

Тема: Знакомство со средой программирования Delphi. Создание консольного приложения.

Задание: научиться основным навыкам работы в среде программирования.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.1. Язык программирования.

Практическое задание № 2.

Тема: Создание синтаксических диаграмм.

Задание: закрепить понятие синтаксических диаграмм.

Проверяемые результаты обучения: ПК 1.2, ПК 3.1

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.2. Константы

Практическое задание № 3.

Тема: Создание программ с использованием констант.

Задание: научиться эффективно использовать константы в языке программирования.

Проверяемые результаты обучения: ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 4.

Тема: Создание программ с применением типизированных констант.

Задание: научиться эффективно использовать типизированные константы и их инициализации в языке программирования.

Проверяемые результаты обучения: ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.3. Выражения

Практическое задание № 5.

Тема: Вычисление арифметических и логических выражений.

Задание: закрепить знания арифметических операций их приоритетов.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.4. Типы данных**Практическое задание № 6.**

Тема: Работа с переменными.

Задание: научить эффективно использовать переменные и типы данных.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 7.

Тема: Применение методов приведения типов данных.

Задание: закрепить основные методы приведения различных типов данных .

Проверяемые результаты обучения: ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.5. Операторы.**Практическое задание № 8.**

Тема: Реализация программы с использованием структурных операторов.

Задание: изучить операторы управления ходом работы программы.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 9.

Тема: Реализация программы с использованием операторов ветвления.

Задание: изучить понятие ветвления.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 10.

Тема: Реализация программы с использованием операторов циклов.

Задание: научить эффективно использовать операторы циклов.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 11.

Тема: Реализация программы с использованием операторов безусловного перехода.

Задание: изучить ошибки при использовании операторов безусловного перехода.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.6. Подпрограммы.**Практическое задание № 12.**

Тема: Создание и использование подпрограмм.

Задание: изучить преимущества процедурного программирования.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 13.

Тема: Передача параметров подпрограммам.

Задание: изучить основные методы передачи параметров подпрограмме.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.7. Структура программы**Практическое задание № 14.**

Тема: Реализация программы с использованием механизма областей видимости **Задание:** изучить понятие областей видимости.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 15.

Тема: Создание примеров «побочного эффекта».

Задание: изучить понятие «побочного эффекта».

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.8. Файловые типы данных**Практическое задание № 16.**

Тема: Чтение и запись информации в текстовые файлы.

Задание: изучить способы записи и чтения текстовых файлов.

Проверяемые результаты обучения: ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 17.

Тема: Чтение и запись информации в двоичные файлы.

Задание: изучить способы записи и чтения двоичных файлов.

Проверяемые результаты обучения: ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.9. Ссылочные типы данных и указатели**Практическое задание № 18.**

Тема: Работа с динамически выделяемой памятью.

Задание: изучить группу понятий: указатель, динамическая память, разыменовывание указателя.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 19.

Тема: Работа с динамическими массивами.

Задание: научить основным методам работы с динамическими массивами.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 2.10. Динамические библиотеки

Практическое задание № 20.

Тема: Создание и использование динамических библиотек. Раннее связывание..

Задание: изучить методы создания и раннего связывания динамических библиотек.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Практическое задание № 21.

Тема: Создание и использование динамических библиотек. Позднее связывание..

Задание: изучить метод позднего связывания динамических библиотек.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 3.3. Способы записи алгоритмов

Практическое задание № 22.

Тема: Построение блок-схем..

Задание: повторить основные элементы блок-схем.

Проверяемые результаты обучения: ПК 1.2, ПК 3.1

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 4.2. Связные списки

Практическое задание № 23.

Тема: Создание связных списков и работа с ними..

Задание: закрепить основные методы работы со связными списками.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 4.3. Графы

Практическое задание № 24.

Тема: Работа с графами

Задание: закрепить основные операции над графами.

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.
За не полностью выполненную работу выставляется - 1 балл.
За невыполненную работу выставляется - 0 баллов.
Время выполнения: 80 минут

Тема 5.1. Внутренняя сортировка (сортировка массивов)

Практическое задание № 25.

Тема: Реализация методов внутренней сортировки данных.
Задание: изучить основные способы сортировки данных в оперативной памяти.
Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.
За не полностью выполненную работу выставляется - 1 балл.
За невыполненную работу выставляется - 0 баллов.
Время выполнения: 80 минут

Тема 5.2. Внешняя сортировка (сортировка файлов).

Практическое задание № 26.

Тема: Реализация методов внешней сортировки.
Задание: изучить основные способы сортировки данных в текстовых файлах.
Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.
За не полностью выполненную работу выставляется - 1 балл.
За невыполненную работу выставляется - 0 баллов.
Время выполнения: 80 минут

Тема 6.1. Поиск в массиве.

Практическое задание № 27.

Тема: Организация поиска в массиве.
Задание: изучить основные методы поиска информации в массиве.
Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.
За не полностью выполненную работу выставляется - 1 балл.
За невыполненную работу выставляется - 0 баллов.
Время выполнения: 80 минут

Тема 6.2. Поиск в таблице.

Практическое задание № 28.

Тема: Использование хеш-функций для поиска в таблице.
Задание: показать преимущество поиска с помощью хэш-функций.
Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.
За не полностью выполненную работу выставляется - 1 балл.
За невыполненную работу выставляется - 0 баллов.
Время выполнения: 80 минут

Тема 7.2. Концепции объекта и класса.

Практическое задание № 29.

Тема: Создание класса Thuman.

Задание: изучить простой пример описания класса.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 7.4. Поля, свойства, методы. События.**Практическое задание № 30.**

Тема: Создание класса Tcircle.

Задание: научить принципу инкапсуляции.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

Тема 7.5. Области видимости в классах. Шаблоны классов. Интерфейсы.**Практическое задание № 31.**

Тема: Создание абстрактного класса Tfigure.

Задание: углубление знаний по объектно-ориентированному программированию.

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 4 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 80 минут

3.1.2 Внеаудиторные самостоятельные работы

Внеаудиторная самостоятельная работа №1. Подготовка конспекта по теме «Развитие программирования как науки».

Проверяемые результаты обучения: ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №2. Подготовка конспекта по теме «Современный рейтинг сред разработки».

Проверяемые результаты обучения: ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №3. Подготовка конспекта по теме «Виды жизненных циклов разработки программного обеспечения».

Проверяемые результаты обучения: ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №4. Подготовка конспекта по теме «История развития среды программирования Delphi».

Проверяемые результаты обучения: ПК 1.1, ПК 1.2

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №5.

Выполнение дифференцированных заданий по теме «Формулы Бэкуса-Наура».

Проверяемые результаты обучения: ПК 1.2, ПК 3.1

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №6.

Выполнение дифференцированных заданий по теме «Типизированные константы».

Проверяемые результаты обучения: ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №7. Выполнение дифференцированных заданий по теме «Построение и вычисление выражений»

Выполнение дифференцированных заданий по теме «Типизированные константы».

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №8: Подготовка конспекта по теме «Преобразование типов»

Проверяемые результаты обучения: ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №9.

Выполнение дифференцированного задания по теме «Теорема о структурном программировании»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №10. Выполнение дифференцированного задания по теме «Способы передачи фактических параметров в подпрограмму»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №11. Выполнение дифференцированного задания по теме «Области действия и видимости декларации»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №12. Выполнение дифференцированного задания по теме «Стандартные подпрограммы работы с файлами»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №13. Выполнение дифференцированного задания по теме «Статические и динамические переменные программы»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №14. Подготовка конспекта по теме «Динамические библиотеки»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №15. Подготовка конспекта по теме «Тезис Тьюринга»

Проверяемые результаты обучения: ПК 1.2, ПК 3.1

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №16. Подготовка конспекта по теме «Основные элементы блок-схем»

Проверяемые результаты обучения: ПК 1.2, ПК 3.1

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №17. Подготовка конспекта по теме «Классификация структур данных»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №18. Подготовка конспекта по теме «Двунаправленный список и методы работы с ним»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №19. Подготовка конспекта по теме «Матрица смежности, матрица инцидентности»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №20. Подготовка конспекта по теме «Пузырьковая сортировка, шейкерная сортировка, сортировка простым выбором»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №21. Подготовка конспекта по теме «Быстрая сортировка»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №22. Подготовка конспекта по теме: «Операции разделения и слияния файлов»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №23. Подготовка конспекта по теме «Использование алгоритмов внутренней сортировки»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №24. Подготовка конспекта по теме «Линейный поиск»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №25. Подготовка конспекта по теме «Хеш-функции»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №26. Подготовка конспекта по теме

«Структурное, процедурное, модульное, функциональное программирование»

Проверяемые результаты обучения: ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №27. Подготовка конспекта по теме «Различие в понятиях объект и класс»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №28. Подготовка конспекта по теме «Иерархия классов. Способы наследования членов класса»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №29. Подготовка конспекта по теме «Основное назначение неявной переменной self и объектной переменной sender»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

Внеаудиторная самостоятельная работа №30. Подготовка конспекта по теме «Директивы видимости: private, public, protected, published, automated»

Проверяемые результаты обучения: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

Критерии оценки:

За верное выполнение работы выставляется - 3 балла.

За не полностью выполненную работу выставляется - 1 балл.

За невыполненную работу выставляется - 0 баллов.

Время выполнения: 120 минут

6. Задания промежуточной аттестации

6.1 Тестовое задание

Раздел 1. Базовые понятия программирования

1. Какими знаками заканчивается большинство строк кода в Си++?
 - 1) **;** (точка с запятой)
 - 2) **,** (запятая)
 - 3) **.** (точка)
 - 4) **:** (двоеточие)
2. Какое значение, по умолчанию, возвращает программа операционной системе в случае успешного завершения?
 - 1) 1
 - 2) Программа не возвращает значение.
 - 3) **0**
 - 4) -1
3. Какая из следующих записей - правильный комментарий в С++?
 - 1) **/* комментарий */**
 - 2) {комментарий}
 - 3) **** комментарий ****
 - 4) ***/ комментарий */**
4. Какая из следующих записей - правильный комментарий в С++?
 - 1) **// комментарий**
 - 2) **/-- комментарий**
 - 3) **/* комментарий**
 - 4) **# комментарий**
5. Какая строка содержит только зарезервированные слова языка программирования С++?
 - 1) **if, else, for, while do, switch, continue, break**
 - 2) defaulted, goto, return, extern, private, public, protected
 - 3) sizeof, const, typedef, static, voided, enum, struct, union
 - 4) char, int, float, doubled, short, long, unsigned, signed
6. Какое подключение заголовочного файла библиотеки будет неверным?
 - 1) **#include<iostream>**
 - 2) **#include "iostream"**
 - 3) **#include <main>**
 - 4) **#include "main"**
7. Какие среды программирования (IDE) предназначены для разработки программных средств?
 - 1) **MS Visual Studio, NetBeans, QT Creator, RAD Studio, Dev-C++**
 - 2) MS Visual Studio, Code::Blocks, QT Creator, RAD Studio, MathCAD
 - 3) MS Visual Studio, Code::Blocks, QT Creator, AutoCAD, Eclipse
8. Какие скобки используются для подключения заголовочных файлов системных библиотек
 - 1) Квадратные
 - 2) Фигурные
 - 3) Круглые
 - 4) **Угловые**
9. Какие скобки определяют блок программного кода
 - 1) Квадратные
 - 2) **Фигурные**
 - 3) Круглые
 - 4) Угловые
10. Какие служебные символы используются для обозначения начала и конца блока кода?
 - 1) **()**
 - 2) **[]**
 - 3) **<>**
 - 4) **{ }**
11. Какой заголовочный файл необходимо подключить, чтобы вызвать функцию isalpha()?
 - 1) **ifstream.h**
 - 2) **cstring**

- 3) conio.h
4) **cctype**
12. Какую функцию должны содержать все программы на C++?
1) program()
2) **main()**
3) start()
4) system()
13. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется:
1) **транслятор**
2) компилятор
3) интерпретатор
4) сканер
14. Укажите объектно-ориентированный язык программирования
1) C++
2) Java
3) **Всевариантыответов**
4) Delphi
15. Чтобы подключить заголовочный файл в программу на C++, например iostream необходимо написать:
1) include #iostream,h;
2) #include<>; с iostream.h внутри скобок
3) include (iostreamh)
4) **#include <> с iostreamвнутрискобок**
16. Язык программирования C++ разработал
1) **БьериСтрауструп**
2) Дональд Кнут
3) Никлаус Вирт
4) Кен Томпсон
17. Что означает std?
1) **Это название пространства имен, в котором находится функция cout**
2) Это название класса, у которого вызывается не статический метод cout
3) Это название класса, у которого вызывается статический метод cout
4) Это название класса, в котором находится функция cout
18. Библиотека, отвечающая за взаимодействие с системой (определяет функцию system()), а также для функции для работы с генератором случайных чисел srand() и rand(), называется:
Ответ: cstdlib
19. Библиотека, отвечающая за работу с математическими функциями, называется:
Ответ: smath
20. Библиотека, отвечающая за работу с потоками ввода-вывода, называется:
Ответ: iostream
21. Директива препроцессора, позволяющая подключать заголовочные файлы определяется ключевым словом:
Ответ: include
22. Какими символами начинается однострочный комментарий в языке C++ - это
Ответ: //
23. Какой файл имеет расширение *.h в языке C++ (1 слово):
Ответ: заголовочный
24. Какой файл имеет расширение *.hpp в языке C++ (1 слово):
Ответ: исходный
25. Название пространства имен, подключаемого в соответствии с новым стандартом языка:
Ответ: std
26. Программа, преобразующая исходный код (*.cpp) на языке C++ в объектный код (*.o)
Ответ: компилятор

27. Программа, преобразующая объектный код (*.o) файлов проекта в исполняемый код (*.exe) Ответ: компоновщик
28. Программа, преобразующая исходный код с учетом всех директив (например #include) Ответ: препроцессор
29. Смысл любого оператора в любом языке программирования определяет...
Ответ: семантика
30. Верно ли, что в языке C++ регистр символов важен при определении переменных?
Верно
Неверно
31. Верно ли, что на языке C++ написано большинство современных операционных систем?
Верно
Неверно

Раздел 2. Основные конструкции языка программирования

32. Как обозначается оператор взятия остатка деления в языке C++:
Ответ: %
33. Как обозначается оператор деления в языке C++:
Ответ: /
34. Как обозначается оператор присваивания в языке C++:
Ответ: =
35. Как обозначается оператор присваивания языка C++, выполняющий вычитание:
Ответ: -=
36. Как обозначается оператор присваивания языка C++, выполняющий деление:
Ответ: /=
37. Как обозначается оператор присваивания языка C++, выполняющий умножение:
Ответ: *=
38. Как обозначается оператор присваивания языка C++, выполняющий сложение:
Ответ: +=
39. Как обозначается оператор сдвига влево в языке C++:
Ответ: <<
40. Как обозначается оператор сдвига вправо в языке C++:
Ответ: >>
41. Какая функция в языке C++ определяет возведение в степень?
Ответ: pow
42. Какая функция в языке C++ определяет квадратный корень:
Ответ: sqrt
43. Какая функция в языке C++ определяет модуль числа (абсолютное значение)?
Ответ: abs
44. Какая функция в языке C++ определяет округление вещественного числа в большую сторону?
Ответ: floor
45. Какой заголовочный файл C++ содержит инструкции файлового ввода/вывода?
1) **fstream**
2) ifstream
3) ifstream
4) iostream
46. Какой из следующих классов обрабатывает процесс записи в файл?
1) **ofstream**
2) другое
3) ifstream
4) input_file
47. Какой из следующих классов обрабатывает процесс чтения из файла?
1) ofstream
2) другое
3) **ifstream**

- 4) `input_file`
48. Как организовать запись в файл?
- 1) `a_file<< "запись";`
 - 2) `a_file="запись";`
 - 3) `a_file.printf("запись");`
 - 4) `a_file.out("запись");`
49. Как организовать считывание из файла?
- 1) `a_file>> "считывание";`
 - 2) `a_file=" считывание ";`
 - 3) `a_file.scanf("запись");`
 - 4) `a_file.in("запись");`
50. Простые типы данных в C++.
- 1) целые - `int`, вещественные - `float` или `real`, символьные - `char`
 - 2) целые - `int`, вещественные - `float` или `double`, символьные - `string`
 - 3) целые - `bool`, вещественные - `float` или `double`, символьные - `string`
 - 4) **целые - `int`, вещественные - `float` или `double`, символьные - `char`**
51. Поставьте в соответствие типам данных их описание
- | | |
|--------------------|--------------|
| <code>long</code> | целый |
| <code>float</code> | вещественный |
| <code>bool</code> | логический |
| <code>char</code> | символьный |
52. Поставьте в соответствие типам данных их описание
- | | |
|---------------------|--------------|
| <code>int</code> | целый |
| <code>double</code> | вещественный |
| <code>bool</code> | логический |
| <code>char</code> | символьный |
53. Укажите последовательность выполнения операторов в цикле `for`, если цикл обозначить следующим образом: `&g(Оператор1; Оператор 2; Оператор 3) Блок_кода`
- | | |
|-----------|---|
| Оператор1 | 1 |
| Оператор2 | 2 |
| Оператор3 | 4 |
| Блок_кода | 3 |
54. В каком случае лучше всего использовать приведение типов данных?
- 1) чтобы изменить тип возвращаемого значения функции
 - 2) чтобы разрешить программе использовать только целые числа
 - 3) **при делении двух целых чисел, для того, чтобы вернуть результат с плавающей точкой**
 - 4) во всех выше указанных случаях
55. Вывод данных в C++
- 1) `cout<< выражение1, выражение2, ...;`
 - 2) `cout>> выражение1 << выражение2 <<endl<< ...;`
 - 3) `cout>> выражение1, выражение2, ...;`
 - 4) `cout<< выражение1 << выражение2 ...;`
56. Ввод данных в C++
- 1) `cin>> выражение1, выражение2, ...;`
 - 2) `cin>> выражение1>>выражение2...;`

- 3) `cin >> выражение1 >> выражение2 >> endl >> ...;`
- 4) `cin << выражение1 << выражение2 ...;`

57. Выберите правильный вариант объявления константной переменной в C++, где `type` - тип данных в C++
`variable` - имя переменной `value` - константное значение

- 1) `constvariable = value;`
- 2) **`const type variable = value;`**
- 3) `consttypevariable := value;`

58. Какое значение будет напечатано?

```

1 int main(int argc, char** argv)
2 {
3     int x = 0;
4     int y = 0;
5
6     if (x++ && y++)
7     {
8         y += 2;
9     }
10
11     std::cout << x + y << std::endl;
12
13     return 0;
14 }
```

- 1) 4
- 2) 2
- 3) 1
- 4) 3

59. К вещественному типу данных относится:

- 1) **float**
- 2) `bool`
- 3) `long`
- 4) `char`

60. К вещественному типу данных относится:

- 1) `longint`
- 2) `int`
- 3) **double**
- 4) `bool`

61. Какое из следующих значений эквивалентно зарезервированному слову `true`?

- 1) 1
- 2) 0.1
- 3) -1
- 4) **Все варианты ответов**
- 5) 66

62. Какое ключевое слово указывает, что целая переменная не может принимать отрицательные значения?

- 1) `positive`
- 2) `long`
- 3) **unsigned**
- 4) нет такого зарезервированного слова

63. К логическому типу данных относится:

- 1) `long`
- 2) `float`
- 3) **bool**
- 4) `double`

64. К логическому типу данных относится:

- 1) `char`
- 2) **bool**
- 3) `int`
- 4) `longint`

65. К символьному типу данных относится:

- 1) `int`
- 2) `float`
- 3) **char**

- 4) longint
66. К символьному типу данных относится:
- 1) **char**
 - 2) long
 - 3) double
 - 4) bool
67. К целому типу данных относятся. Выберите один или несколько ответов:
- 1) **int**
 - 2) **long**
 - 3) float
 - 4) char
68. К целому типу данных относятся. Выберите один или несколько ответов:
- 1) char
 - 2) double
 - 3) **long int**
 - 4) bool
69. Какие преобразования типов данных не возможны без потери данных?
- 1) char ->float
 - 2) все перечисленные преобразования не возможны
 - 3) int ->float
 - 4) **float ->int**
70. Какие преобразования типов данных не возможны без потери данных?
- 1) int -> char
 - 2) float -> double
 - 3) **double -> float**
 - 4) все перечисленные преобразования не возможны
71. Какие преобразования типов данных не возможны без потери данных?
- 1) double ->int
 - 2) float ->int
 - 3) double -> float
 - 4) **все перечисленные преобразования не возможны**
72. Какие преобразования типов данных не возможны без потери данных?
- 1) int -> char
 - 2) float -> double
 - 3) int ->float
 - 4) **нет ответа**
73. Какой заголовочный файл следует подключить, чтобы можно было пользоваться приведением типов данных?
- 1) cctype
 - 2) **Никакого**
 - 3) cmath
 - 4) iostream
74. Какой из перечисленных типов данных не является типом данных в C++?
- 1) int
 - 2) float
 - 3) double
 - 4) **const**
75. Объявлена переменная char a; Какое из следующих выражений не верно?
- 1) a = '3';
 - 2) **a = "3";**
 - 3) a = 3;
 - 4) a = (3);
76. Оператор присваивания в C++:
- 1) =
 - 2) :=
 - 3) ==

4) ==

77. Оператор вывода cout может печатать несколько значений или переменных в одной команде, используя следующий синтаксис:

- 1) cout<< ("Привет" & name & "n");
- 2) cout<< "Привет", name, "n";
- 3) cout<< "Привет" + name + "n";
- 4) **cout<< "Привет" <<name<< "n";**

78. Оператор ввода cin может принимать несколько значений переменных в одной команде, используя следующий синтаксис:

- 1) cin>> "Введите var1" >>var1 >>"Введите var2" >>var2;
- 2) **cin>> var1 >> var2;**
- 3) cin>> "Введите var1" + var1 >>"Введите var2" + var2;
- 4) cin<< var1 << var2;

79. Почему приведение типов данных может быть не безопасно.

- 1) **Вы можете временно потерять часть данных - таких, как отсечение десятичной части чисел с плавающей точкой.**
- 2) Некоторые преобразования не определены компилятором, такие как - преобразование символа в целое.
- 3) Вы можете навсегда изменить значение переменной.
- 4) нет никаких опасностей.

80. Преобразование целочисленной переменной в вещественный тип данных:

- 1) float a = 56;
- 2) float a = float (56);
- 3) float a = static_cast<float>56;
- 4) Все варианты ответов

81. Выберите вариант, где используется явное преобразование:

- 1) float a = 56;
- 2) float a = float (56);
- 3) **float a = static_cast<float>56;**
- 4) 56/1.0;

82. Выберите вариант, где используется неявное преобразование:

- 1) float a = 56;
- 2) float a = float (56);
- 3) 56/1.0;
- 4) **Все варианты ответов**

83. Преобразование целочисленной переменной value в ASCII эквивалент

- 1) **(char) value**
- 2) cout<< value
- 3) atoi(value)
- 4) char (value)

84. Результат выполнения следующего фрагмента кода: 54 << 3?

- 1) 440
- 2) 556
- 3) 623
- 4) **432**

85. Результат выполнения следующего фрагмента кода: 168 >> 3?

- 1) 56
- 2) **21**
- 3) 504
- 4) 1344

86. Результат выполнения следующего фрагмента кода: cout<< 22 / 5 * 3;

- 1) **12**
- 2) 1.47
- 3) 13.2
- 4) 13

87. Результат выполнения следующего фрагмента кода: cout<< 22 * 3 / 5;

- 1) 12
- 2) **1.47**

3) 13.2

4) **13**

88. Укажите верные способы объявить переменную `a` вещественного типа данных, равную 0. Выберите один или несколько ответов:

1) **`float a=0;`**

2) **`float a(0);`**

3) `int a = 0;`

4) `int a(0);`

89. Укажите верные способы объявить переменную `a` целочисленного типа данных, равную 0. Выберите один или несколько ответов:

1) `float a=0;`

2) `float a(0);`

3) **`int a = 0;`**

4) **`int a(0);`**

90. Укажите операцию, приоритет выполнения которой больше остальных

1) `+`

2) `()`

3) `++`

4) `*`

91. Укажите операцию, приоритет выполнения которой больше остальных

1) `+`

2) `++`

3) `*`

4) `/`

92. Укажите правильное приведение типа данных

1) `char:a;`

2) `a(char);`

3) `to(char, a);`

4) **`(char)a;`**

93. Какое значение примет переменная `c` в результате выполнения следующей операции: `c = (a = 3, b = 4)` Ответ: 4

94. Какое значение примет переменная `c` в результате выполнения следующей операции: `c = (a = 3, b = 4, d = 18)`

Ответ: 18

95. Какое количество байт занимает тип данных `bool` в 32х разрядной операционной системе

Ответ: 1

96. Какое количество байт занимает тип данных `char` в 32х разрядной операционной системе

Ответ: 1

97. Какое количество байт занимает тип данных `double` в 32х разрядной операционной системе

Ответ: 8

98. Какое количество байт занимает тип данных `float` в 32х разрядной операционной системе

Ответ: 4

99. Какое количество байт занимает тип данных `int` в 32х разрядной операционной системе

Ответ: 4

100. Какое количество байт занимает тип данных `long` в 32х разрядной операционной системе

Ответ: 4

101. Какое количество байт занимает тип данных `wchar_t` в 32х разрядной операционной системе

Ответ: 2

102. Чему будет равно значение `c` в следующем выражении: `double c = 4 / 3;`

1) **1**

2) 2

3) Возникнет ошибка компиляции

4) 1.333333

103. Какое значение вернет следующее выражение ($a = 3, b = 4$):
 Ответ: 4
104. Какое значение вернет следующее выражение ($a = 3, b = 4, c = 5, d = 6$):
 Ответ: 6
105. Какое значение вернет следующее выражение: $a = b = c = 10$;
 Ответ: 10
106. Чему в результате вычисления будет равно значение переменной a , если $a = 1/4$ Ответ: 0
107. Чему в результате вычисления будет равно значение переменной a , если $a = 3/4$ Ответ: 0
108. Чему в результате вычисления будет равно значение переменной a , если $a = 9/4$ Ответ: 2
109. Чему равен результат выполнения следующего выражения: $1000 / 100 \% 7 * 2$
- 1) 250
 - 2) 1000
 - 3) **6**
 - 4) 10
110. Что будет напечатано, после выполнения этого кода: `cout<< (5 << 3);` ?
- 1) 53
 - 2) **40**
 - 3) 35
 - 4) 15
111. Что будет напечатано, после выполнения этого кода: `cout<< (120 >> 3);` ?
- 5) 53
 - 6) 40
 - 7) 35
 - 8) **15**
112. Это значение $5.9875e17$ может быть сохранено в переменной, типа:
- 1) short
 - 2) **float**
 - 3) int
 - 4) long
113. Это значение $5.9875e39$ может быть сохранено в переменной, типа:
- 1) float
 - 2) int
 - 3) long
 - 4) **double**
114. Операция декремента (уменьшение значения на единицу) в языке C++ записывается: Ответ: -
115. Операция инкремента (увеличение значения на единицу) в языке C++ записывается: Ответ: ++
116. Поток ввода в языке C++
 Ответит
117. Поток вывода в языке C++
 Ответ: cout
118. Пустой тип данных в языке программирования C++ записывается как:
 Ответ: void

Раздел 3. Теория алгоритмов

119. В каком случае можно не использовать фигурные скобочки в операторе выбора if?
- 1) если в теле оператора if нет ни одного оператора
 - 2) нет правильного ответа
 - 3) если в теле оператора if два и более операторов
 - 4) **если в теле оператора if всего один оператор**
120. Выберите верный алгоритм поиска максимума
- 1) **int a[10];
int max = a[0];

for(int i = 1; i < 10; i++)
{**

```

if(a[i] > max ) max = a[i];
}

```

- 2) int a[10];
int max = a[0];

```

for(int i = 1; i < 10; i++)
{
if(a[i] == max )
max = a[i];
}

```

- 3) int a[10];
int max = a[0];

```

for(int i = 1; i < 10; i++)
{
if(a[i] < max )
max = a[i];
}

```

121. Выберите верный алгоритм поиска минимума

- 1) int a[10];
int min = a[0];

```

for(int i = 1; i < 10; i++)
{
if(a[i] > min )
max = a[i];
}

```

- 2) int a[10];
int min = a[0];

```

for(int i = 1; i < 10; i++)
{
if(a[i] == min )
max = a[i];
}

```

- 3) **int a[10];**
int max = a[0];

```

for(int i = 1; i < 10; i++)
{
if(a[i] < min )
max = a[i];
}

```

122. Если условие оператора выбора if ложное, то:

- 1) **выполняется следующий оператор, сразу после оператора if**
- 2) программа завершает работу
- 3) выполняется тело оператора выбора if

123. Какое значение вернет следующее выражение (true или false):

(!(2<3) || (3>4) || !(4<7))

Ответ: false

124. Какое значение вернет следующее выражение (true или false): (!(2<3)||(3>4))

Ответ: false

125. Какое значение вернет следующее выражение (true или false):

(!(2<3)&&(3>4))

Ответ: true

126. Какое значение вернет следующее выражение (true или false):

`((2<3)||(3>4)&&(4<7))`

Ответ: true

127. Какое значение вернет следующее выражение (true или false):

`((2<3)&&!(3>4)&&(4<7))`

Ответ: true

128. Какое значение вернет следующее выражение (true или false):

`((2<3)||(3>4))`

Ответ: true

129. Какое значение вернет следующее выражение (true или false):

`((2<3)&&(3>4))`

Ответ: false

130. Какое значение вернет следующее выражение (true или false):

`((2<3),(3>4))`

Ответ: false

131. Какое значение вернет следующее выражение (true или false): `((2<3)||!(3>4))`

Ответ: true

132. Какое значение вернет следующее выражение (true или false): `((2<3)&&!(3>4))`

Ответ: true

133. Каков будет результат выражения `!(1 && !(0 || 1))`?

- 1) false
- 2) неоднозначность
- 3) **true**

134. Результат выполнения следующего фрагмента кода: `!(1 || 0) && 0`

- 1) **1**
- 2) результат не может быть заранее определен
- 3) 0
- 4) Другое

135. Результат выполнения следующего фрагмента кода: `!(1 && 0) && 0`

- 1) **1**
- 2) результат не может быть заранее определен
- 3) 0
- 4) Другое

136. Результат выполнения следующего фрагмента кода: `!(1 || 0) || 0`

- 1) 1
- 2) результат не может быть заранее определен
- 3) **0**
- 4) Другое

137. Какие скобки определяют условия

- 1) Квадратные
- 2) Фигурные
- 3) **Круглые**
- 4) Угловые

138. Какой из ниже перечисленных вариантов ответа, показывает правильно записанный оператор выбора if

- 1) `if{ условное выражение }`
- 2) **`if(условное выражение)`**
- 3) условное выражение `if`
- 4) `if` условное выражение

139. Какой из ниже перечисленных операторов, не является циклом в C++?

- 1) `dowhile`
- 2) **`switch`**
- 3) `while`
- 4) `for`

140. Какой из следующих логических операторов - логический оператор И?

- 1) **И**
- 2) **`&&`**
- 3) `I`
- 4) `&`

141. Какой из следующих логических операторов - логический оператор ИЛИ?
 1) **И**
 2) &&
 3) I
 4) &
142. Какой из следующих логических операторов - логический оператор НЕ?
 1) -
 2) **!**
 3) !!
 4) no
143. Какой из следующих операторов - оператор сравнения двух переменных?
 1) :=
 2) ==
 3) =
 4) ===
144. Ключевое слово, обозначающее начало оператора цикла с предусловием:
 Ответ: while
145. Ключевое слово, обозначающее начало условного оператора:
 Ответ: if
146. Ключевое слово, обозначающее начало цикла с постусловием:
 Ответ: do while
147. Ключевое слово, обозначающее начало цикла с параметром:
 Ответ: for
148. Логический оператор И в языке C++
 Ответ: &&
149. Логический оператор ИЛИ в языке C++
 Ответ: И
150. Логический оператор НЕ в языке C++
 Ответ: !
151. Логическое значение ИСТИНА в языке C++ обозначается как
 Ответ: true
152. Логическое значение ЛОЖЬ в языке C++ обозначается как
 Ответ: false
153. Оператор сравнения НЕРАВНО в языке C++
 Ответ: !=
154. Оператор сравнения РАВНО в языке C++
 Ответ: ==
155. Какой оператор не допускает перехода от одного константного выражения к другому?
 1) stop;
 2) **break;**
 3) end;
 4) точка с запятой
156. Логическая операция с большим приоритетом выполнения.
 1) **!**
 2) &&
 3) ||
 4) *
157. Оператор if else позволяет определить действие ...
 1) только для ложного условия
 2) только для истинного условия
 3) **для истинного и ложного условий**
 4) другое
158. Тело любого цикла выполняется до тех пор, пока его условие ...
 1) ложно
 2) у цикла нет условия

- 3) **истинно**
- 4) в любом случае

159. Тело оператора if, будет выполняться, если его условие:

- 1) ложно (false)
- 2) **истинно (true)**

160. Укажите блок кода, в котором переменная u доступна

```

1 int main(int argc, char** argv)
2 {
3
4     if ( argc > 10 )
5     {
6     }
7     else if (int y = argc - 1 )
8     {
9
10    }
11    else
12    {
13
14    }
15
16    return 0;
17 }
```

- 1) строки 8-11
- 2) строки 4-17
- 3) строки 4-15
- 4) **строки 8 -14**

161. Укажите верно написанный оператор if:

- 1) **if (условие) {Оператор;}**
- 2) if (условие) {Оператор}
- 3) if условие {Оператор;}
- 4) if (условие); {Оператор}
- 5) if (условие) {Оператор};

162. Укажите верно написанный оператор while:

- 1) **while (условие) {Оператор;}**
- 2) while (условие) {Оператор}
- 3) while условие {Оператор;}
- 4) while (условие); {Оператор}
- 5) while (условие) {Оператор};

163. Укажите верный условный оператор, выводящий на экран true или false в зависимости от значения логического выражения.

- 1) if a<b
cout<< "true";
else
cout<< "false";
- 2) **if (a<b) cout<< "true"; else
cout<< "false";**
- 3) if (a<b) cout<< "true" else
cout<< "false";
- 4) if (a<b); cout<< "true"; else
cout<< "false";

- 1) <=
- 2) >=
- 3)
- 4)**

- 1)

```
1 // форма записи оператора цикла do while:
2 do // начало цикла do while
3 {
4 /*блок операторов*/
```
- 2)

```
1 // форма записи оператора цикла do while:
2 do // начало цикла do while
3 { _____ :
4 _____
```
- 3)

```
1 // форма записи оператора цикла do while:
2 do // начало цикла do while
3 {
4 /*блок операторов*/;
5 }
6 while {/*условие выполнения цикла*/}; // конец цикла do while
```

166. Цикл с параметром, это?

- 1) do while
- 2) while

- 3) **for**
- 4) switch

164. Укажите неправильно записанную операцию отношения все операторы записаны правильно =!

165. Укажите правильную форму записи цикла dowhile

167. Цикл с постусловием, это?

- 1) **do while**
- 2) while
- 3) for
- 4) switch
- 5)

168. Цикл с предусловием, это?

- 1) do while
- 2) **while**
- 3) for
- 4) switch

169. Какому зарезервированному слову программа передаёт управление в случае, если значение переменной или выражения оператора switch не совпадает ни с одним константным выражением?

- 1) other
- 2) **default**
- 3) contingency
- 4) all

170. Что появится на экране, после выполнения этого фрагмента кода:

```
1 int a = 1, b = 2;
2 if (a == b);
3 cout << a << " = " << b << endl;
```

1) a = b
2) синтаксическая ошибка
3) вывод на экран не выполнится
4) **1 = 2**

171. Оператор множественного выбора, это?

- 1) multiple
- 2) case
- 3) if
- 4) **switch**

172. Что будет напечатано, после выполнения следующего фрагмента кода:

```
1 int main()
2 {
3     for (int i = 0; i < 4; ++i)
4     {
5         switch (i)
6         {
7             case 0 : std::cout << "0";
8             case 1 : std::cout << "1"; continue
9             case 2 : std::cout << "2"; break;
10            default : std::cout << "D"; break;
11        }
12        std::cout << "
13    }
14    return 0;
15 }
```

- 1) Ошибка компиляции в строке 10
- 2) 01.2.D.
- 3) 0.1.2.
- 4) **0112.D.**
- 5) 011.2.D

173. Каков результат работы следующего фрагмента кода?

```
1 int x = 0;
2
3     switch(x)
4     {
5
6         case 1: cout << "Один";
7
8         case 6: cout << "Ноль";
9
10        case 2: cout << "Привет мир";
11
12    }
```

- 1) Привет мир
- 2) Один
- 3) Ноль
- 4) **НольПривет мир**

174. Какое значение будет напечатано, в результате выполнения программы?

```

1 #include <iostream>
2
3 int main()
4 {
5     int x = 3;
6
7     switch(x)
8     {
9         case 6:
10            int x = 1;
11            std::cout << x << std::endl;
12            break;
13        case 3:
14            std::cout << x << std::endl;
15            break;
16        default:
17            x = 2;
18            std::cout << x << std::endl;
19    }
20    return 0;
21 }

```

- 1) 1
- 2) 2
- 3) ничего не напечатается, программа вообще не будет работать
- 4) **3**
- 5) 0

175. В каком фрагменте кода будут напечатаны 20 звездочек - *?

- 1) inti, N = 20;
for(i = 20; i < N; i--) cout << "*";
- 2) inti, N = 20;
for(i = 19; i < N; i--) cout << "*";
- 3) **inti, N = 20;**
for(i = 0; i < N; N--) cout << "*";
- 4) inti, N = 40; for(i = 0; i < N; i--) cout << "*";

176. Какой служебный знак ставится после оператора case ?

- 1) -
- 2) .
- 3) **:**
- 4) ;

177. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf) for(i = 0; i <= 4; i++)

Блок_кода

Ответ: 5

178. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf) for(i = 0; i <= 1024; i++)

Блок_кода

Ответ: 1025

179. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf) for(i = 0; i < 1024; i++)

Блок_кода

Ответ: 1024

180. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf) for(i = 0; i < 4; i++) Блок_кода

Ответ: 4

181. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf) for(i = 0; i == 4; i++)

Блок_кода

Ответ: 0

182. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf)

for(i = 0; i == 1024; i++) Блок_кода

Ответ: 0

183. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf)

for(i = 0; i <= 4; i--) Блок_кода;

Ответ: inf

184. Сколько раз выполнится блок кода в цикле for? (бесконечность обозначать inf)

for(i = 0; i < 4; i--) Блок_кода;

Ответ: inf

185. Вид алгоритма, в котором в зависимости от условия операторы языка выполняются по нескольким независимым сценариям, называется:

Ответ: разветвляющийся

186. Вид алгоритма, в котором в зависимости от условия операторы языка выполняются несколько раз, называется:

Ответ: циклический

187. Вид алгоритма, в котором операторы языка выполняются последовательно сверху вниз без ответвлений и циклов, называется:

Ответ: линейный

188. Чему будет равна переменная a, после выполнения этого кода: int a; for(a = 0; a < 10; a++) {}

- 1) 10
- 2) 1
- 3) **9**
- 4) 0

189. Переменная x может быть доступна в другом блоке программы?

```

1 int main(int argc, char** argv)
2 {
3     if (argc > 2)
4     {
5         int x = 5;
6     } else {
7
8     }
9     return 0;
10
11
12
13 }
```

Верно

Неверно

190. Может ли переменная x быть доступна в блоке else оператора if?

Верно

Неверно

Раздел 4. Структуры данных

191. В каком из следующих вариантов ответов выполнен корректный доступ к полю var структуры b? (Структура объявлена статически)

- 1) var.b
- 2) b-var;
- 3) var-b;
- 4) **b.var;**

192. В каком из следующих вариантов ответов выполнен корректный доступ к полю var структуры b? (Структура объявлена динамически)

- 1) b.var;
- 2) var->b
- 3) **b->var;**
- 4) b>var;

193. Правильное объявление переменной, типа структуры foo

- 1) foo;
- 2) struct foo;
- 3) int foo;
- 4) **foovar;**

194. Правильное определение структуры в C++

- 1) structa_structint a;
- 2) **structa_struct {int a};**

- 3) structa_struct {int a;}
- 4) struct {int a;}

195. При определении структуры необходимо использовать следующее ключевое слово

- 1) record
- 2) **struct**
- 3) object
- 4) structure

196. Укажите правильный доступ к переменной структуры

- 1) **b.var;**
- 2) b-var;
- 3) b->var;
- 4) b>var;

197. Будет ли напечатано сообщение не равны?

```

1 struct Foo
2 {
3 };
4 struct Bar
5 {
6 };
7
8
9
10 int main(int argc, char** argv)
11 {
12     Foo* f = new Foo;
13     Bar* b = new Bar;
14
15     if (f == b)
16         std::cout << "равны" << std::endl;
17     else
18         std::cout << "не равны" << std::endl;
19
20     return 0;
21 }

```

Верно

Неверно

Раздел 5. Сортировка информации

198. Массив - это ...

- 1) Массив - это упорядоченные в памяти элементы одного и того же типа, имеющие имя. Доступ к отдельным элементам массива осуществляется по имени массива и адресу
- 2) **Массив - это упорядоченные в памяти элементы одного и того же типа, имеющие имя. Доступ к отдельным элементам массива осуществляется по имени массива и индексу**
- 3) Массив - это упорядоченные в памяти элементы одного и того же типа, имеющие общий адрес. Доступ к отдельным элементам массива осуществляется по адресу и индексу

199. В какой из следующих строк выполняется обращение к седьмому элементу массива?

- 1) mas+7;
- 2) mas(7);
- 3) **mas[6];**
- 4) mas[7];
- 5) mas(6);

200. В каком из вариантов ответов объявлен двумерный массив целых чисел?

- 1) chararray[20];
- 2) arrayanarray[20][20];
- 3) intarray[20, 20];
- 4) **intanarray [20][20];**

201. Выберите верный цикл for, который выводит на экран числа в диапазоне от 0 до 9

- 1) for(inti = 0; i < 9; i++)
 - {
 - cout<<i<<endl;
 - }
- 2) **for(inti = 0; i < 10; i++)**
 - {
 - cout<<i<<endl;
 - }
- 3) for(inti = 1; i < 10; i++);
 - {

```
cout<<i<<endl;
}
```

- 4) for(int i = 1; i < 9; i++);
 {
 cout<<i<<endl;
 }

202. До каких пор будут выполняться операторы в теле цикла while (x < 100)?

- 1) Пока x больше ста
- 2) **Пока x строго меньше ста**
- 3) Пока x равен ста
- 4) Пока x меньше или равен ста

203. Какое значение будет напечатано, в результате выполнения следующего кода?

```
1
2
3 int fnain(void)
4 {
5     int sum = 0;
6
7     int array[3][3] = {{0, 1, 2}, {3, 4, 5}, {6, 7, 8}};
8
9     for (int i = 0; i < 3; ++i)
10    {
11        for (int j = 2; j < 3; j++)
12        {
13            sum += array[i][j];
14        }
15    }
16
17    std::cout << sum << std::endl;
18
19    return 0;
20
```

- 1) **15**
- 2) синтаксическая ошибка
- 3) 21
- 4) 9

204. Какие служебные символы используются для работы с индексом?

- 1) ()
- 2) []
- 3) <>
- 4) { }

205. Какой порядковый номер последнего элемента массива, размер массива 19?

- 1) 19
- 2) порядковый номер определяется программистом
- 3) **18**
- 4) 20

206. Какие скобки используются для работы с массивом

- 1) **Квадратные**
- 2) Фигурные
- 3) Круглые
- 4) Угловые

207. Укажите правильное объявление массива

- 1) arrayanarray[10];
- 2) anarray{10};
- 3) **intanarray[10];**
- 4) intanarray;

208. Укажите строку, которая возвращает адрес первого элемента в массиве arr:

- 1) **arr[0];**
- 2) arr[1];

- 3) arr;
- 4) &arr;

Раздел 6. Поиск информации

209. Допустим у нас есть массив `char arr[8]`; И в массив `arr` мы попытались с помощью оператора `cin` записать следующий набор символов `HelloWorld`. Что в действительности будет содержать массив `arr`?
- 1) **HelloWo**
 - 2) Hello W
 - 3) HelloWorld
 - 4) Hello
210. Каким символом завершается Си-строка?
- 1) "
 - 2) \n
 - 3) '
 - 4) ;
211. Словосочетание "Helloworld!" может быть сохранено в символьном массиве размером `n` элементов. Укажите чему равно `n`?
- 1) 10
 - 2) 11
 - 3) 12
 - 4) **13**
212. Строковый типы данных в C++
- 1) строки в C++ представляются как массивы элементов типа `char`, заканчивающиеся терминатором строки - символом с нулевым значением `'0'`.
 - 2) строки в C++ представляются как массивы элементов типа `char`, заканчивающиеся терминатором строки - символом с нулевым значением `(\0)`.
 - 3) **строки в C++ представляются как массивы элементов типа `char`, заканчивающиеся терминатором строки - символом с нулевым значением `(\0)`.**
213. Укажите корректное определение строковой переменной. Выберите один или несколько ответов:
- 1) `string[20] mystr;`
 - 2) `string mystr[20];`
 - 3) **`string mystr;`**
 - 4) **`char mystr[20];`**
214. Укажите литерал (константу) строки в языке C++
- 1) **"Статическая строка"**
 - 2) 'Статическая строка'
 - 3) `charstring[100];`
 - 4) `constcharstring;`
215. Какая из следующих функций считывает 100 символов из входного потока в строку `x`?
- 1) `read(x);`
 - 2) `cin.getline(100, x, 'n');`
 - 3) `readline(x, 100, 'n');`
 - 4) **`cin.getline(x, 100, 'n');`**
216. Какая из следующих функций добавляет одну строку в конец другой?
- 1) **`strcat ()`**;
 - 2) `strcpy();`
 - 3) `strncpy();`
 - 4) `strlen ();`
217. Какая из следующих функций копирует полностью одну строку в другую?
- 1) `strcat ();`
 - 2) **`strcpy();`**
 - 3) `strncpy();`
 - 4) `strlen ();`
218. Какая из следующих функций копирует фиксированное количество символов из одной строки в другую?
- 1) `strcat ();`
 - 2) `strcpy();`

- 3) **strncpy()**;
 4) `strlen ()`;
219. Какая из следующих функций определяет размер строки?
 1) `strcat ()`;
 2) `strcpy()`;
 3) `strncpy()`;
 4) **`strlen ()`**;
220. Какая из следующих функций ищет символ в строке?
 1) `strcmp ()`;
 2) `strfind()`;
 3) **`strchr ()`**;
 4) `strlen ()`;
221. Какая из следующих функций сравнивает две строки?
 1) `compare()`;
 2) `stringcompare()`;
 3) **`strcmp()`**;
 4) `cmp()`;
222. Каков результат работы следующей программы?

```
char s[] = "HeIoMIII";
_____ ccut << strlen(s) << " " << sizeof s;
```

- 1) **5 9**
 2) 7 20
 3) 5 20
 4) 8 20

Раздел 7. Объектно-ориентированное программирование (ООП).

223. Выберите верный способ объявить функцию в языке C++:
 1) `тип_имя_функции, параметры`;
 2) `параметры имя_функции (тип_результата)`;
 3) **`тип_результатаимя_функции(параметры)`**;
 4) `имя_функции параметры (тип_результата)`;
224. Правильное определение функции в C++
 1) **`void name { }`**
 2) `funct name { }`
 3) `void name { };`
 4) `funct name { };`
225. В какой из следующих записей используется операция взятия адреса?
 1) `*a`;
 2) **`&a`**;
 3) `address(a)`;

4) a&;

226. Выберите правильное (полное) определение функции

1)

```
1 int funct(int x)
2 {
3     return x = x + 1;
4 }
```

```
1 int funct();
```

2)

3)

```
1 void funct(int)
2 {
3     tout « "Hello"
4 }
```

227. Укажите правильный вызов функции. Предполагается, что функция была объявлена ранее.

- 1) **funct();**
- 2) funct;
- 3) intfunct();
- 4) funct x, y;

228. Указанный код объявляет массив ссылок: int&x[50];

Верно

Неверно

229. Поименованный набор описаний и операторов, выполняющих определенную задачу:

Ответ: функция

230. Можно ли гарантировать, что объявленная встроенная функция действительно является встроенной?

Верно

Неверно231. **Можно ли прегрузить функцию шаш()?**

Верно

Неверно

```
1 #include <iostream>
2
3 using namespace std;
4
5 int foo(int y);
6 int foo(int x)
7 {
8     return x+1;
9 }
10
11 int main(int argc, char** argv)
12 {
13     int x = 3;
14     int y = 6;
15
16 cout << foo(x) << endl;
17
18 return 0;
19
```

232. Что будет напечатано на экране, после выполнения этого кода?

- 1) 3
- 2) ошибка компиляции
- 3) **4**
- 4) 9

233. Укажите тип возвращаемого значения следующей функции: intfunc(char x, float v, double t);

- 1) **int**
- 2) float
- 3) double
- 4) char
- 5) Функция не возвращаемого типа

234. Укажите тип возвращаемого значения следующей функции: voidfunc(char x, float v, double t);

- 1) void
- 2) float
- 3) double
- 4) char
- 5) **Функция не возвращаемого типа**

235. Для чего используются встроенные функции?

- 1) Для удаления ненужных функций
- 2) Чтобы уменьшить размер программы
- 3) Для упрощения файла с исходным кодом

4) Для увеличения скорости работы программы

236. Какие из следующих функций являются встроенными?
- 1) **inlinevoidfoo() {}**
 - 2) нет правильного ответа
 - 3) inline:void foo() {}
 - 4) void foo() inline {}
237. Что значит ключевое слово inline?
- 1) препроцессор хранит в памяти код функции
 - 2) **все вызовы встроенных функции заменяются кодом этой функции**
 - 3) сообщает компилятору использовать функцию только в пределах одного файла с исходным кодом
 - 4) нет правильного ответа
238. Какая из следующих записей возвращает значение переменной a, хранящееся в памяти по адресу на который указывает указатель?
- 1) &a;
 - 2) *a;
 - 3) val(a);
 - 4) a
239. Какие из следующих функций являются встроенными?
- 1) **inlinevoidfoo() {}**
 - 2) нет правильного ответа
 - 3) voidfoo() inline {}
 - 4) inline:voidfoo() {}
240. Какие скобки определяют получение параметров в функциях
- 1) **Круглые**
 - 2) Квадратные
 - 3) Фигурные
 - 4) Угловые
241. Корректное выделение памяти
- 1) int *a = new 20;
 - 2) int a = new sizeof(int * 20);
 - 3) **int *a = new int[20];**
 - 4) int a = new int(20);
242. Корректное выделение памяти
- 1) int a = new int[20];
 - 2) int *a = new int(20);
 - 3) int *a = new sizeof(int * 20);
 - 4) **int *a = new int;**
243. Какие скобки используются для работы с индексом и для выделения памяти
- 1) **Квадратные**
 - 2) Фигурные
 - 3) Круглые
 - 4) Угловые
244. После выполнения ряда операций с указателем, что будет выведено на экран?

```

1 int main(int argc, char** argv)
2 {
3 // предположим, int занимает 4 байта
4 cout << sizeof(int) << endl;
5
6 int *x = new int;
7
8 // предположим адрес равен 0x60450009
9 cout << x << endl;
10
11 cout << x + 3 << endl;
12
13 return 0;
14 }

```

- 1) некорректное определение
- 2) нельзя заранее сказать, каково будет значение адреса

- 3) **0x60450003**
 4) 0x60450000
 5) 0x6045000c
245. Укажите правильное объявление указателя в C++
 1) ptr x;
 2) int&x;
 3) **int *x;**
 4) int x;
246. Укажите зарезервированное ключевое слово для динамического выделения памяти?
 1) value
 2) **new**
 3) malloc
 4) create
247. Укажите зарезервированное ключевое слово для высвобождения выделенной памяти!
 1) clear
 2) remove
 3) free
 4) **delete**
248. Строка Привет Мир будет показана на экране или нет?

```

1 int main(int argc, char** argv)
2 {
3     int array[33];
4
5     if ( &array[4] < &array[23] )
6
7         std::cout << "Привет мир" << std::endl;
8     }
9
10    return 0;
11 }
```

- 1) синтаксическая ошибка
 2) нет
 3) **да**
249. Что из нижеперечисленного не является прототипом функции?
 1) void funct();
 2) intfunct(char x, char y);
 3) **double funct(char x)**
 4) char x();
250. Что такое ссылка?
 1) **ссылка является псевдонимом для имени переменной**
 2) оператор
 3) используется для переименования объектов
 4) нет правильного ответа

Время выполнения: 40 минут

Перечень объектов контроля и оценки

Наименование объектов контроля и оценки	Основные показатели оценки результата	Оценка (кол-во баллов)
ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	Электронный тест	20 баллов
ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.		
ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.		
ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.		
ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.		
ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.		
ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.		
ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.		
ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.		
ПК 1.1. Выполнять разработку спецификаций отдельных компонент.		
ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых		
ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.		
ПК 1.4. Выполнять тестирование программных модулей.		
ПК 1.5. Осуществлять оптимизацию программного кода модуля.		
ПК 3.1. Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.		

4. Шкала оценки образовательных достижений

Процент результативности (правильных ответов)	Качественная оценка индивидуальных образовательных достижений	
	балл (отметка)	вербальный аналог
91 - 100	5	отлично
76 -90	4	хорошо
61 -75	3	удовлетворительно
менее 61	2	неудовлетворительно
более 61	зачтено	
менее 61	не зачтено	