

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ИНКЛЮЗИВНОГО ВЫСШЕГО ОБРАЗОВАНИЯ

**«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО  
ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Факультет Прикладная математика и информатика  
Кафедра Информационных технологий и прикладной математики

«Утверждаю»

Зав. кафедрой 

«26» августа 2020 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ  
ПО ДИСЦИПЛИНЕ**

Программирование на ЭВМ

образовательная программа направления подготовки  
01.03.02 "Прикладная математика и информатика"  
Б1.В.ДВ.07.01 «Дисциплины (модули)», часть, формируемая участниками  
образовательных отношений, дисциплины (модули) по выбору

Профиль подготовки  
Вычислительная математика и информационные технологии

Квалификация (степень) выпускника  
Бакалавр

Форма обучения: очная

Курс 2 семестр 3

Москва  
2020

Составитель / составители: МГГЭУ, доцент кафедры информационных технологий и прикладной математики

место работы, занимаемая должность

  
подпись

Белоглазов А.А.  
Ф.И.О.

«22» августа 2020 г.  
Дата

Рецензент: МГГЭУ, профессор кафедры информационных технологий и прикладной математики

место работы, занимаемая должность

  
подпись

Истомина Т.В.  
Ф.И.О.

«23» августа 2020 г.  
Дата

Согласовано:

*Представитель работодателя или объединения работодателей*

Генеральный директор, АО «Микропроцессорные системы», к.т.н.

(должность, место работы)

  
подпись

Демидов Л.Н. «26» августа 2020 г.  
Ф.И.О. Дата

Фонд оценочных средств рассмотрен и одобрен на заседании кафедры Информационных технологий и прикладной математики (протокол № 1 от «26» августа 2020 г.)

/Зав. кафедрой ИТиПМ/

  
подпись

Петрунина Е.В. «26» августа 2020 г.  
Ф.И.О. Дата

Дополнения и изменения, внесенные в фонд оценочных средств, утверждены на заседании кафедры \_\_\_\_\_,

протокол № \_\_\_\_\_ от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой \_\_\_\_\_ / Ф.И.О/

Дополнения и изменения, внесенные в фонд оценочных средств, утверждены на заседании кафедры \_\_\_\_\_,

протокол № \_\_\_\_\_ от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой \_\_\_\_\_ / Ф.И.О/

Дополнения и изменения, внесенные в фонд оценочных средств, утверждены на заседании кафедры \_\_\_\_\_,

протокол № \_\_\_\_\_ от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой \_\_\_\_\_ / Ф.И.О/

## Содержание

1. Паспорт фонда оценочных средств.....
2. Перечень оценочных средств.....
3. Описание показателей и критериев оценивания компетенций.....
4. Методические материалы, определяющие процедуры оценивания результатов обучения, характеризующих этапы формирования компетенций.....
5. Материалы для проведения текущего контроля и промежуточной аттестации.....
- ...

## 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине «Программирование на ЭВМ»

Оценочные средства составляются в соответствии с рабочей программой дисциплины и представляют собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.), предназначенных для измерения уровня достижения обучающимися установленных результатов обучения.

Оценочные средства используются при проведении текущего контроля успеваемости и промежуточной аттестации.

Таблица 1 - Перечень компетенций, формируемых в процессе освоения дисциплины

<b>Код и содержание компетенции</b>	<b>Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций</b>
ПК-1. Способен собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям	ПК-1.1. Знает передовые научные достижения в области своих научных интересов; основные методы и средства сбора, алгоритмы обработки и интерпретации данных современных научных исследований.
	ПК-1.2. Умеет систематизировать научные результаты, выделять из них главное, и удалять второстепенное; объективно оценивать результаты научных разработок, выполненных другими специалистами; самостоятельно выбирать эффективные методы решения поставленных задач.
	ПК-1.3. Владеет методами, приемами, алгоритмами и способами сбора, обработки и интерпретации данных; данными современных научных исследований, необходимых для формирования выводов по соответствующим научным исследованиям; навыками формирования выводов по соответствующим научным исследованиям.
ПК-2. Способен понимать, совершенствовать и применять современный математический аппарат	ПК-2.1. Знает основные теоремы и формулы математического анализа, геометрии, дискретной математики, дифференциальных уравнений, теоретических основ информатики, численных методов, функционального анализа.
	ПК-2.2. Умеет применять основные теоремы и формулы математического анализа, геометрии, дискретной математики, дифференциальных уравнений, теоретических основ информатики, численных методов.
	Владеет методами, приемами, алгоритмами и способами применения современного математического аппарата для решения задач профессиональной деятельности.
ПК-7. Способен к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения	Знает теоретические основы разработки программных и алгоритмических решений в области системного и прикладного программного обеспечения; математические методы решения задач, процедурный и объектно-ориентированный подходы к разработке информационных систем; актуальные проблемы в области программирования; методы и технологии программирования; языки программирования, основы

	технологии модульного программирования на языках высокого уровня.
	Умеет применить математический метод для решения задачи; подобрать рациональную технологию программирования для решения профессиональной задачи; создавать программные продукты и алгоритмические решения в области системного и прикладного программного обеспечения.
	Владеет навыками применения математических методов для решения задач и применения стандартных алгоритмов; навыками разработки и создания алгоритмических и программных решений в области системного и прикладного программного обеспечения; навыками разработки программных приложений с использованием современных языков программирования.

Конечными результатами освоения дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование дескрипторов происходит в течение всего семестра по этапам в рамках контактной работы, включающей различные виды занятий и самостоятельной работы, с применением различных форм и методов обучения (табл.2).

Таблица 2 - Формирование компетенций в процессе изучения дисциплины:

Код компетенции	Уровень освоения компетенций	Индикаторы достижения компетенций	Вид учебных занятий <sup>1</sup> , работы, формы и методы обучения, способствующие формированию и развитию компетенций <sup>2</sup>	Контролируемые разделы и темы дисциплины <sup>3</sup>	Оценочные средства, используемые для оценки уровня сформированности компетенции <sup>4</sup>
ПК-1 ПК-2		<i>Знает</i>			
	Недостаточный уровень	Студент не способен разрабатывать, внедрять и адаптировать прикладное программное обеспечение. Не знает характеристик применения ПО; методов разработки, анализа и проектирования ПО, структурного программирования и объектно-ориентированного программирования, основы визуального программирования, программирования с использованием компонентов. обработки событий и реакции на событие (процедура-обработчик события), на языке высокого уровня C++.	Лекционные и практические занятия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам
	Базовый	Студент усвоил основное	Лекционные и	Тема 1. Основы	Текущий контроль –

<sup>1</sup> Лекционные занятия, практические занятия, лабораторные занятия, самостоятельная работа...

<sup>2</sup> Необходимо указать активные и интерактивные методы обучения (например, интерактивная лекция, работа в малых группах, методы мозгового штурма и т.д.), способствующие развитию у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств.

<sup>3</sup> Наименование темы (раздела) берется из рабочей программы дисциплины.

<sup>4</sup> Оценочное средство должно выбираться с учетом запланированных результатов освоения дисциплины, например:

«Знать» – собеседование, коллоквиум, тест...

«Уметь», «Владеть» – индивидуальный или групповой проект, кейс-задача, деловая (ролевая)

игра, портфолио...

	уровень	содержание материала дисциплины, но имеет несистематизированные знания о теоретических основах информатики, знание и понимание функциональных характеристик применения ПО; методов разработки, анализа и проектирования ПО, структурного программирования и объектно-ориентированного программирования, основы визуального программирования, программирования с использованием компонентов. обработки событий и реакции на событие (процедура-обработчик события), на языке высокого уровня C++.	практические занятия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	устный опрос, защита отчетов по практическим работам
	Средний уровень	Студент способен самостоятельно выделять главные положения в изученном материале. Знает теоретические основы информатики, характеристики применения ПО; методов разработки, анализа и проектирования ПО, структурного программирования и объектно-ориентированного программирования, основы визуального программирования, программирования с использованием компонентов. обработки событий и реакции на	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам

		событие (процедура-обработчик события), на языке высокого уровня C++.			
Высокий уровень		Студент знает, понимает, выделяет главные положения в изученном материале и способен самостоятельно использовать теоретические основы информатики. Показывает глубокое знание и понимание функциональных характеристик применения ПО; методов разработки, анализа и проектирования ПО, структурного программирования и объектно-ориентированного программирования, основы визуального программирования, программирования с использованием компонентов. обработки событий и реакции на событие (процедура-обработчик события), на языке высокого уровня C++.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам
		<i>Умеет</i>			
Базовый уровень		Студент умеет работать с инструментальными средствами, поддерживающими создание ПО для ИС MS Visual Studio2017.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное	Текущий контроль – устный опрос, защита отчетов по практическим работам



			аттестации.	проектирование приложений	
Средний уровень	Студент в основном умеет работать с инструментальными средствами, поддерживающими создание ПО для ИС MS Visual Studio2017	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Высокий уровень	Студент умеет на высоком уровне, работать с инструментальными средствами, поддерживающими создание ПО для ИС MS Visual Studio2017	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
	<i>Владеет</i>				
Базовый уровень	Студент владеет базовыми навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и реализации программного продукта MS Visual Studio2017.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	

	Средний уровень	Студент на среднем уровне владеет навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и реализации программного продукта MS Visual Studio2017.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам
	Высокий уровень	Студент владеет навыками самостоятельного применения современного математического аппарата для решения задач профессиональной деятельности и может применить их на практике. Студент владеет навыками самостоятельного определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и реализации программного продукта MS Visual Studio2017.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам
ПК-7		<i>Знает</i>			
	Недостаточный уровень	Студент не способен проектировать ИС в соответствии с профилем подготовки по видам обеспечения. Не знает методы разработки, анализа и проектирования ПО; функциональное и техническое	Лекционные и практические занятия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования	Текущий контроль – устный опрос, защита отчетов по практическим работам

		проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание ПО.		Тема 4. Визуальное проектирование приложений	
Базовый уровень	Студент имеет несистематизированные знания о методах разработки, анализа и проектирование ПО; функционального и технического проектирования; паттернов проектирования; номенклатуры инструментальных средств, поддерживающих создание ПО.	Лекционные и практические занятия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Средний уровень	Студент знает основное содержание материала дисциплины. Знает методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание ПО.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Высокий уровень	Студент знает, понимает, выделяет главные положения в изученном материале и способен дать краткую характеристику основным идеям проработанного материала дисциплины. Знает методы разработки, анализа и проектирования ПО; функциональное и техническое	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование	Текущий контроль – устный опрос, защита отчетов по практическим работам	

		проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание ПО.		приложений	
		<i>Умеет</i>			
Базовый уровень	Студент имеет затруднения при использовании нотации построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Средний уровень	Студент в основном умеет использовать нотации для построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Высокий уровень	Студент умеет самостоятельно использовать нотации для построения функциональной и процессной моделей исследуемой предметной области.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное	Текущий контроль – устный опрос, защита отчетов по практическим работам	

			аттестации.	Проектирование приложений	
		<i>Владеет</i>			
Базовый уровень	Студент на базовом уровне владеет навыками проектирования прикладных программных продуктов.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Средний уровень	Студент владеет знаниями всего изученного материала. Владеет навыками проектирования прикладных программных продуктов.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	
Высокий уровень	Студент владеет концептуально-понятийным аппаратом. Владеет навыками проектирования прикладных программных продуктов.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации.	Тема 1. Основы программирования Тема 2. Процедурное программирование Тема 3. Основы объектно-ориентированного программирования Тема 4. Визуальное проектирование приложений	Текущий контроль – устный опрос, защита отчетов по практическим работам	

## 2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ<sup>5</sup>

Таблица 3

№	Наименование оценочного средства	Характеристика оценочного средства	Представление оценочного средства в ФОС
1.	Опрос	Средство контроля усвоения учебного материала темы, раздела или разделов дисциплины, организованное как учебное занятие в виде собеседования преподавателя с обучающимися.	Вопросы по темам/разделам дисциплины
2.	Защита отчетов по практическим работам; или решение разноуровневых задач (заданий)	<p>Различают задачи и задания:</p> <p>а) репродуктивного уровня, позволяющие оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и умение правильно использовать специальные термины и понятия, узнавание объектов изучения в рамках определенного раздела дисциплины;</p> <p>б) реконструктивного уровня, позволяющие оценивать и диагностировать умения синтезировать, анализировать, обобщать фактический и теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей;</p> <p>в) творческого уровня, позволяющие оценивать и диагностировать умения, интегрировать знания различных областей, аргументировать собственную точку зрения.</p>	Комплект разноуровневых задач (заданий)

<sup>5</sup> Указываются оценочные средства, применяемые в ходе реализации рабочей программы данной дисциплины.

3.	Тестирование	Средство, позволяющее оценить уровень знаний обучающегося путем выбора им одного из нескольких вариантов ответов на поставленный вопрос. Возможно использование тестовых вопросов, предусматривающих ввод обучающимся короткого и однозначного ответа на поставленный вопрос.	Тестовые задания
4.	Экзамен		Вопросы к экзамену

### 3. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Оценивание результатов обучения по дисциплине «Практикум программирования на ЭВМ» осуществляется в соответствии с Положением о текущем контроле успеваемости и промежуточной аттестации обучающихся.

Предусмотрены следующие виды контроля: текущий контроль (осуществление контроля всех видов аудиторной и внеаудиторной деятельности обучающегося с целью получения первичной информации о ходе усвоения отдельных элементов содержания дисциплины) и промежуточная аттестация (оценивается уровень и качество подготовки по дисциплине в целом).

Показатели и критерии оценивания компетенций, формируемых в процессе освоения данной дисциплины, описаны в табл. 4.

Таблица 4.

Код компетенции	Уровень освоения компетенции	Индикаторы достижения компетенции	Критерии оценивания результатов обучения
ПК-1 ПК-2 ПК-7		Знает	
	Недостаточный уровень Оценка «незачтено».	ПК-1.1. ПК-2.1. ПК-7.1.	<i>Не знает значительной части материала курса, не способен самостоятельно выделять главные положения в изученном материале дисциплины.</i>
	Базовый уровень Оценка «зачтено».	ПК-1.1. ПК-2.1. ПК-7.1.	<i>Знает не менее 50 % основного материала курса, однако испытывает затруднения в его применении.</i>
	Средний уровень Оценка «зачтено».	ПК-1.1. ПК-2.1. ПК-7.1.	<i>Знает основную часть материала курса, способен применить изученный материал на практике, испытывает незначительные затруднения в решении задач.</i>
	Высокий уровень Оценка «зачтено».	ПК-1.1. ПК-2.1. ПК-7.1.	<i>Показывает глубокое знание и понимание материала, способен применить изученный материал на практике.</i>
		Умеет	
	Базовый уровень	ПК-1.2.	<i>Умеет воспроизвести не менее 50 % основного материала курса, однако</i>



		<i>ПК-2.2. ПК-7.2.</i>	<i>испытывает затруднения при решении практических задач.</i>
Средний уровень		<i>ПК-1.2. ПК-2.2. ПК-7.2.</i>	<i>Умеет решать стандартные профессиональные задачи с применением полученных знаний, испытывает незначительные затруднения в решении задач.</i>
Высокий уровень		<i>ПК-1.2. ПК-2.2. ПК-7.2.</i>	<i>Умеет решать стандартные профессиональные задачи с применением полученных знаний, показывает глубокое знание и понимание материала, способен решить задачу при изменении формулировки.</i>
		Владеет	
Базовый уровень		<i>ПК-1.3. ПК-2.3. ПК-7.3.</i>	<i>Студент владеет основными навыками теоретического и практического применения методов аналитической геометрии, линейной алгебры и математического анализа. Имеет несистематизированные знания основных разделов дисциплины.</i>
Средний уровень		<i>ПК-1.3. ПК-2.3. ПК-7.3.</i>	<i>Студент владеет знаниями всего изученного материала, владеет навыками теоретического и практического применения методов аналитической геометрии, линейной алгебры и математического анализа. Испытывает незначительные затруднения в решении задач.</i>
Высокий уровень		<i>ПК-1.3. ПК-2.3. ПК-7.3.</i>	<i>Свободно владеет навыками теоретического и практического применения методов аналитической геометрии, линейной алгебры и математического анализа, показывает глубокое знание и понимание изученного материала. Студент владеет концептуально-понятийным аппаратом, научным языком и терминологией профессиональной деятельности.</i>

## **4. Методические материалы, определяющие процедуры оценивания результатов обучения**

### **Задания в форме устного опроса:**

Устный опрос используется для текущего контроля успеваемости обучающихся по дисциплине в качестве проверки результатов освоения материала. Каждому студенту выдается свой собственный, узко сформулированный вопрос. Ответ должен быть четким и кратким, содержащим все основные характеристики описываемого понятия. В своем ответе студент должен показать умения проследивать причинно-следственные связи и навыки рассуждений и доказательства.

### **Задания в форме практических работ. Комплект разноуровневых задач (заданий)**

Практическая работа представляет собой контрольное мероприятие по учебному материалу каждой темы (раздела) дисциплины, состоящее в индивидуальном выполнении обучающимся практических заданий для оценки полученных знаний, умений и владений компетенциями, формируемыми по данной дисциплине.

Выполнение практических работ является средством текущего контроля успеваемости обучающихся по дисциплине и может включать в себя следующие типы заданий: задания типового вида и задания творческого характера, по результатам выполнения практических заданий обучающие оформляют отчеты, содержащие анализ полученных результатов и выводы.

### **Задания в форме тестирования**

Тест представляет собой контрольное мероприятие по учебному материалу каждой темы (раздела) дисциплины, состоящее в выполнении обучающимся системы стандартизированных заданий, которая позволяет автоматизировать процедуру измерения уровня знаний и умений обучающегося.

Тестирование является средством текущего контроля успеваемости обучающихся по дисциплине и может включать в себя следующие типы заданий: задание с единственным выбором ответа из предложенных вариантов, задание на определение верных и неверных суждений; задание с множественным выбором ответов.

В каждом задании необходимо выбрать все правильные ответы.

## **5. Материалы для проведения текущего контроля и промежуточной аттестации**

### **Задания в форме устного опроса**

#### **Тема 1. Основы программирования**

- 1) Назначение, состав и структура программного обеспечения.
- 2) Организация взаимодействия пользователя с ЭВМ.
- 3) Обработка программ под управлением ОС.
- 4) Обобщенная структура операционной системы.
- 5) Краткая характеристика современных операционных систем.
- 6) Общая характеристика языков программирования, области их применения.

Компиляторы интерпретаторы.

- 7) Системы программирования. Технологии разработки алгоритмов и приложений.
- 8) Основные этапы разработки приложений. Определение алгоритма.
- 9) Свойства алгоритма. Способы описания алгоритмов: словесный, схемный, с помощью псевдокода или языка программирования.
- 10) Единая система программной документации (ЕСПД): содержание, вид, форма.
- 11) Методы разработки алгоритмов и программ: нисходящее, восходящее.
- 12) Модульное представление программ. Структурное программирование.
- 13) Объектно-ориентированная технология.
- 14) Тестирование и отладка приложений. Методы тестирования. Типы ошибок.
- 15) Способы и средства обнаружения и локализации синтаксических и логических ошибок.
- 16) Организация отладки и тестирования приложений.

## **Тема 2. Процедурное программирование**

- 1) Программирование на языке программирования высокого уровня.
- 2) Элементы языка: алфавит, идентификаторы, константы, выражения, операции, встроенные математические функции.
- 3) Приоритеты операций. Структура программы.
- 4) Определение констант и типов данных, объявление переменных и меток.
- 5) Приведение типов и функции преобразования типов.
- 6) Операторы. Инструкции ввода вывода данных.
- 7) Форматирование выводимой информации.
- 8) Правила разработки приложений.
- 9) Организация программ линейной структуры.
- 10) Программирование алгоритмов разветвляющейся и циклической структуры.
- 11) Одномерные и многомерные статические массивы. Динамические массивы.
- 12) Обработка текстовой информации.
- 13) Способы представления текстов. Символы и строки.
- 14) Встроенные подпрограммы обработки строк.
- 15) Подпрограммы. Механизмы передачи параметров в подпрограммы.
- 16) Локальные и глобальные параметры. Область видимости и время жизни переменной.
- 17) Побочные эффекты функций и процедур.
- 18) Математическая рекурсия, рекурсивные подпрограммы. Текстовые и типизированные файлы.
- 19) Прямой и последовательный доступ.

## **Тема 3. Основы объектно-ориентированного программирования**

- 1) Тип данных класс.
- 2) Составляющие класса: поля методы, одноименные методы, свойства.
- 3) Объявление класса. Объект.
- 4) Основные понятия: инкапсуляция, наследование.
- 5) Полиморфизм и виртуальные методы.
- 6) Конструкторы и деструкторы.

## **Тема 4. Визуальное проектирование приложений**

- 1) Визуальное проектирование приложений.
- 2) Особенности функционирования операционной системы Windows.
- 3) Принцип событийного управления.

- 4) Реализация принципов ООП в интегрированной среде разработки.
- 5) Этапы создания приложения. Основы визуального программирования.
- 6) Иерархия классов. Форма и ее модификация.
- 7) Изменение свойств формы.
- 8) Программирование с использованием компонентов.
- 9) Библиотека визуальных компонентов. Объекты и их свойства.
- 10) События и реакции на событие (процедура-обработчик события), методы.
- 11) События связанные с фокусом для формы и компонентов.
- 12) Графический инструментарий.
- 13) Основные понятие: холст, карандаш и кисть.
- 14) Методы реализации графических примитивов.
- 15) Базовые операции преобразования изображений: перемещение, масштабирование, поворот.
- 16) Анимация. Использование битовых образов.
- 17) Перемещение изображения по сложному фону.
- 18) Взаимодействие битового образа с фоном.
- 19) Приложения с длительным циклом.
- 20) Компоненты: линейка, движок, статусная строка, флажок, радиогруппа, радиокнопка, таймер.
- 21) Стандартный диалог. Приложение с несколькими формами.

Контролируемые компетенции: ПК-1, ПК-2, ПК-7

Оценка компетенций осуществляется в соответствии с таблицей 4.

### Задания в форме практических работ. Комплект разноуровневых задач (заданий)

#### Вариант 1

1. Составить программу вычисления суммы и произведения четных чисел из промежутка от 1 до 10.
2. Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{4x^2 + 1}{x - 5}, & \text{если } x < 5 \\ 3x^2 - 2, & \text{если } x \geq 5. \end{cases}$$

#### Вариант 2

1. Составьте программу вывода на экран всех нечетных трехзначных чисел.6. Написать регулярные
2. Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{5x^2 + 2}{x + 4}, & \text{если } x > -4, \\ 3x^2 + 7, & \text{если } x \leq -4 \end{cases}$$

#### Вариант 3

1. Дано целое число N(больше 0). Найти сумму  $1 + 1/2 + 1/3 \dots 1/N$ (вещественное число).
2. Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{7x^2 - 1}{2x + 6}, & \text{если } x < -3, \\ 4x^2 - 5, & \text{если } x \geq -3 \end{cases}$$

## Контролируемые компетенции: ПК-2, ПК-7

Оценка компетенций осуществляется в соответствии с таблицей 4.

### Задания в форме тестирования

#### Вариант 1

- 1 Выберите правильную форму описания строки s на языке Паскаль.
  1. var s:length(s);
  2. var s:set of char;
  3. var s:string;
  4. type s of string;
- 2 Выберите правильную форму описания множества, состоящего из целых чисел.
  1. var M:set of integer;
  2. var K:set of char;
  3. var D:set of string;
  4. var T:set of byte;
- 3 Функция, определяющая длину строки s.
  1. length(s);
  2. concat(s);
  3. pos(s);
  4. insert(s);
- 4 Назначение функции CHR(N).
  1. Определяет по символу порядковый номер
  2. Определяет по порядковому номеру символ
  3. Объявляет символьный тип данных
  4. Выдает номер первого вхождения символа N в строке s
- 5 Назначение функции ORD(C).
  1. Определяет по символу порядковый номер
  2. Определяет по порядковому номеру символ
  3. Объявляет символьный тип данных
  4. Выдает номер первого вхождения символа N в строке s
- 6 Назначение функции POS(s1,s).
  1. Вставляет подстроку s1 в строку s
  2. Удаляет из строки s подстроку s1
  3. Ищет первое вхождение подстроки s1 в строке s
  4. Сцепляет строки s1,s
- 7 Назначение функции Concat(s1,s).
  1. Вставляет подстроку s1 в строку s
  2. Удаляет из строки s подстроку s1
  3. Ищет первое вхождение подстроки s1 в строке s
  4. Сцепляет строки s1,s
- 8 Назначение функции Str(x,s).
  1. Вставляет символ x в строку s
  2. Удаляет из строки s символ x
  3. Ищет первое вхождение подстроки x в строке s
  4. Преобразует числовое значение x в строку s
- 9 Назначение функции Val(s,x,error).
  1. Преобразует строковое значение s в числовое x
  2. Вставляет символ x в строку s
  3. Удаляет из строки s символ x
  4. Преобразует числовое значение x в строку s
- 10 Определите результат выполнения операции [1,2,5,9]\*[1,3,5,7]
  1. [1,2,3,5,7,9]

2. [1,2,3]
  3. [1,5]
  4. [2,9]
- 11 Определите результат выполнения операции  $[1,2,5,9]+[1,3,5,7]$
1. [1,2,3,5,7,9]
  2. [1,2,3]
  3. [1,5]
  4. [2,9]
- 12 Определите результат выполнения операции  $[1,2,5,9]-[1,3,5,7]$
1. [1,2,3,5,7,9]
  2. [1,2,3]
  3. [1,5]
  4. [2,9]
- 13 Определите результат выполнения операции 'a' in ['a','b','c','d']
1. false
  2. true
  3. ['a']
  4. ['a','b','c','d']
- 14 Операция in, применяемая в Паскале ко множественному типу, используется для...
1. включения нового элемента во множество
  2. исключения элемента из множества
  3. проверки принадлежности элемента множеству
  4. объединения двух множеств
- 15 Определите результат выполнения операции 7 in [1..5]
1. false
  2. true
  3. [1..5,7]
  4. [1..7]
- 16 Что выведется на экран в результате выполнения следующего фрагмента программы?
- ```
S:='оса';  
s1:=copy(s,1,2);  
k:='новатор';  
delete(k,5,3);  
k1:=concat(s1,k);  
write(k1);
```
1. основа
  2. осор
  3. новатороса
  4. аватар
- 17 Что выведется на экран в результате выполнения следующего фрагмента программы?
- ```
S:='ЧЭМК';  
L:=length(s);  
write(L);
```
1. 6
  2. 5
  3. 4
  4. 3
- 18 Что выведется на экран в результате выполнения следующего фрагмента программы?
- ```
S:='HELLO';  
k:=pos('L',s);  
write(k);
```
1. 1

2. 2
3. 3
4. 4

19 Для получения из строки s:='формальность' строки s:='форма' необходимо использовать команду...

1. insert(s,1,5);
2. copy(s,5,1);
3. delete(s,6,7);
4. delete(s,1,5);

20 Для получения из строки s:='огород' строки s:='город' необходимо использовать команду...

1. delete(s,1,5);
2. delete(s,1,1);
3. delete(s,2,6);
4. delete(s,1,2);

|     |   |
|-----|---|
| 1.  | 3 |
| 2.  | 4 |
| 3.  | 1 |
| 4.  | 2 |
| 5.  | 1 |
| 6.  | 3 |
| 7.  | 4 |
| 8.  | 4 |
| 9.  | 1 |
| 10. | 3 |
| 11. | 1 |
| 12. | 4 |
| 13. | 2 |
| 14. | 3 |
| 15. | 1 |
| 16. | 1 |
| 17. | 1 |
| 18. | 3 |
| 19. | 3 |
| 20. | 2 |

Контролируемые компетенции: ПК-1, ПК-2, ПК-7

Оценка компетенций осуществляется в соответствии с таблицей 4.

### Вопросы к экзамену

1 Структура программы на С++ Описание типов, подключение библиотек. Макроопределение.

2 Использование библиотек для ввода/вывода данных. Функции ввода/вывода стандартного языка С++. Поточный ввод/вывод. Спецификации для ввода/вывода данных.

3 Локальные и глобальные переменные, их описание.

4 Простейшие типы данных С++ операции логические арифметические используемые в С++.

5 Операторы ветвления Примеры их использования Переключатель С++ switch. Правила использования операторов ветвления.

6 Организация циклов Принудительное прекращение цикла. Вложенность циклов; Условия, задаваемые в операторах цикла.

- 7 Функции. Определения функций, прототипы функций, их использование.
- 8 Область видимости и класс памяти.
- 9 Произвольные или производные типы. Тип VOID.
- 10 Структура и массивы. Объединение.
- 11 Указатели и модели памяти,
- 12 Строки, библиотечные функции для работы с ними Символы, библиотечные функции для работы с ними. Стандартные алгоритмы для обработки строк.
- 13 Указатели как формальные параметры. Указатели как возвращаемые значения. Ссылки и использование ссылок. Указатели и многомерные массивы.
- 14 Текстовые и бинарные файлы. Стандартные потоки, используемые при работе с файлами. Режимы работы с файлами. Файловый указатель. Стандартные значения файловых указателей. Операции ввода/вывода в файл/из файла.
- 15 Описание и использование пользовательских типов.
- 16 Динамические массивы. Описание , инициализация, доступ к элементам массива.
- 17 Создание многофайловых проектов. Создание и использование заголовочных файлов.
- 18 Работа в графическом режиме. Основные графические примитивы. Окна вывода в графическом режиме. Создание анимаций. Макроопределения
- 19 Визуальное проектирование приложений. Особенности функционирования операционной системы Windows. Принцип событийного управления. Реализация принципов ООП в интегрированной среде разработки. Этапы создания приложения. Основы визуального программирования.
- 20 Иерархия классов. Форма и ее модификация. Изменение свойств формы. Программирование с использованием компонентов.
- 21 Библиотека визуальных компонентов. Объекты и их свойства. События и реакции на событие (процедура-обработчик события), методы. События, связанные с фокусом для формы и компонентов.
- 22 Графический инструментарий. Основные понятия: холст, карандаш и кисть. Методы реализации графических примитивов.
- 23 Базовые операции преобразования изображений: перемещение, масштабирование, поворот.
- 24 Анимация. Использование битовых образов. Перемещение изображения по сложному фону. Взаимодействие битового образа с фоном.
- 25 Компоненты: линейка, движок, статусная строка, флажок, радиогруппа, радиокнопка, таймер. Стандартный диалог.
- 26 Компоненты: линейка, движок, статусная строка, флажок, радиогруппа, радиокнопка, таймер.
- 27 Стандартный диалог. Приложение с несколькими формами.

**Контролируемые компетенции: ПК-1, ПК-2, ПК-7**

*Оценка компетенций осуществляется в соответствии с таблицей 4.*