

Автономная некоммерческая организация высшего образования
«МОСКОВСКИЙ МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ»

Рабочая программа дисциплины

Технологии и методы программирования

<i>Направление подготовки</i>	Бизнес-информатика
<i>Код</i>	38.03.05
<i>Направленность(профиль)</i>	Информационные системы и технологии в бизнесе
<i>Квалификация выпускника</i>	бакалавр

Москва
2024

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Общепрофессиональные		ОПК-3

2. Компетенции и индикаторы их достижения

Компетенция	Индикаторы достижения компетенций	Планируемые результаты обучения по дисциплине
ОПК-3	Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации	ОПК-3.1 Знает понятие, виды и особенности продуктов и услуг в сфере ИКТ; основы алгоритмизации, современные методологии разработки программных средств; этапы разработки программных средств; методы обеспечения информационной безопасности. ОПК-3.2 Умеет разрабатывать алгоритмы и программы для практической реализации продуктов и услуг в сфере ИКТ. ОПК-3.3 Владеет методами управления процессами создания и использования продуктов и услуг в сфере ИКТ, в частности, навыками разработки алгоритмов и программ для их практической реализации.

3. Описание планируемых результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки).

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ОПК-3		
	- об эволюции и тенденциях развития технологий разработки программного обеспечения;	- использовать критерии технологичности ПО, методы и средства ее повышения; - определять состав и	- навыками использования программных средств для решения практических задач; - навыками программ

	<ul style="list-style-type: none"> - о средствах автоматизации разработки программного обеспечения (CASE-технологии); - об объектном подходе к спецификации, проектированию и тестированию программного обеспечения. - базовые алгоритмы обработки данных; - основные технологии и методы программирования. - этапы и модели жизненного цикла программного обеспечения (ПО); - основные показатели качества ПО 	<p>порядок определения требований к программному обеспечению;</p> <ul style="list-style-type: none"> - самостоятельно выбирать необходимые технологии программирования для решения конкретных задач; - осуществлять организацию процессов тестирования и отладки ПО, методы структурного и функционального тестирования; - использовать состав и принципы составления программной документации. 	<p>ирования на языке высокого уровня;</p> <ul style="list-style-type: none"> - навыками составления технического задания на разработку ПО; - практическими навыками применения приемов разработки программного обеспечения.
--	--	--	---

4. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина относится к обязательной части учебного плана.

Данная дисциплина взаимосвязана с другими дисциплинами, такими как «Информатика», «Информационная безопасность», «Операционные системы», «Компьютерная графика и мультимедиа».

Изучение дисциплины позволит обучающимся реализовывать компетенции в профессиональной деятельности.

В рамках освоения программы бакалавриата выпускники готовятся к решению задач профессиональной деятельности следующих типов: проектный, организационно-управленческий.

Профиль (направленность) программы установлена путем ее ориентации на сферу профессиональной деятельности выпускников.

5. Объем дисциплины

Виды учебной работы	Формы обучения
	очная форма
Общая трудоемкость: зачетные единицы/часы	8/288
Контактная работа:	
Занятия лекционного типа	56
Занятия семинарского типа	92
Промежуточная аттестация: зачет, зачет с оценкой, экзамен	18,25
Самостоятельная работа (СРС)	121,75

6. Содержание дисциплины (модуля), структурированное по темам / разделам с указанием отведенного на них количества академических часов и видов учебных занятий

6.1. Распределение часов по разделам/темам и видам работы

6.1.1. Очная форма обучения

№ п/ п	Раздел/тема	Виды учебной работы (в часах)			
		Аудиторная работа			Самостоятельная работа
		ЛЗ	ПЗ	ЛабЗ	
Раздел 1					
1.	Технология программирования. Основные понятия и подходы	4	4	-	8
2.	Приёмы обеспечения технологичности программных продуктов	4	4	-	8
3.	Определение требований к программному обеспечению и исходных данных для его проектирования	6	6	-	8
4.	Анализ требований и определение спецификаций программного обеспечения при структурном подходе	6	6	-	7,9
	Промежуточная аттестация	0,1			
	Всего по разделу 1	20	20	-	31,9
Раздел 2					
5.	Проектирование программного обеспечения при структурном подходе	4	9	-	12
6.	Анализ требований и определение спецификаций программного обеспечения при объектном подходе	4	8	-	14
7.	Проектирование программного обеспечения при объектном подходе	4	9	-	14
8.	Разработка пользовательских интерфейсов	6	10	-	13,85
	Промежуточная аттестация	0,15			
	Всего по разделу 2	18	36	-	53,85
Раздел 3					
9.	Пример разработки приложения Windows «Записная книжка»	4	8	-	9
10.	Тестирование программных продуктов	4	8	-	9
11.	Отладка программного обеспечения	4	10	-	9
12.	Составление программной документации	6	10	-	9
	Промежуточная аттестация	18			
	Всего по разделу 3	18	36	-	36
	Итого	56	92	-	121,75
	Промежуточная аттестация	18,25			

6.2. Программа дисциплины, структурированная по темам / разделам

6.2.1. Содержание лекционного курса

№	Наименование темы	Содержание лекционного курса
---	-------------------	------------------------------

п/п	(раздела) дисциплины	
1.	Технология программирования. Основные понятия и подходы	Технология программирования и основные этапы ее развития Проблемы разработки сложных программных систем Блочный-иерархический подход к созданию сложных систем Жизненный цикл и этапы разработки программного обеспечения Эволюция моделей жизненного цикла программного обеспечения Ускорение разработки программного обеспечения. Технология RAD Оценка качества процессов создания программного обеспечения
2.	Приёмы обеспечения технологичности программных продуктов	Понятие технологичности программного обеспечения Модули и их свойства Нисходящая и восходящая разработка программного обеспечения Структурное и «неструктурное» программирование. Средства описания структурных алгоритмов Стиль оформления программы Эффективность и технологичность Программирование «с защитой от ошибок» Сквозной структурный контроль
3.	Определение требований к программному обеспечению и исходных данных для его проектирования	Классификация программных продуктов по функциональному признаку Основные эксплуатационные требования к программным продуктам Предпроектные исследования предметной области Разработка технического задания Принципиальные решения начальных этапов проектирования
4.	Анализ требований и определение спецификаций программного обеспечения при структурном подходе	Функциональные диаграммы Диаграммы потоков данных Структуры данных и диаграммы отношений компонентов данных Математические модели задач, разработка или выбор методов решения
5.	Проектирование программного обеспечения при структурном подходе	Диаграммы переходов состояний Функциональные диаграммы Диаграммы потоков данных Структуры данных и диаграммы отношений компонентов данных Математические модели задач, разработка или выбор методов решения
6.	Анализ требований и определение спецификаций программного обеспечения при объектном подходе	UML стандартный язык описания разработки программных продуктов с использованием объектного подхода Определение «вариантов использования» Построение концептуальной модели предметной области Описание поведения. Системные события и операции
7.	Проектирование программного обеспечения при объектном подходе	Определение отношений между объектами Уточнение отношений классов Проектирование классов Компоновка программных компонентов Проектирование размещения программных

		компонентов для распределенных программных систем Особенность спиральной модели разработки. Реорганизация проекта
8.	Разработка пользовательских интерфейсов	Психофизические особенности человека, связанные с восприятием, запоминанием и обработкой информации Пользовательская и программная модели интерфейса Классификации диалогов и общие принципы их разработки Основные компоненты графических пользовательских интерфейсов Реализация диалогов в графическом пользовательском интерфейсе Пользовательские интерфейсы прямого манипулирования и их проектирование Интеллектуальные элементы пользовательских интерфейсов
9.	Пример разработки приложения Windows «Записная книжка»	Разработка технического задания Анализ предметной области, уточнение спецификаций и разработка структурной схемы Проектирование интерфейса пользователя Проектирование классов приложения
10.	Тестирование программных продуктов	Виды контроля качества разрабатываемого программного обеспечения Ручной контроль программного обеспечения Структурное тестирование Функциональное тестирование Тестирование модулей и комплексное тестирование Оценочное тестирование
11.	Отладка программного обеспечения	Отладка программного обеспечения Классификация ошибок Методы отладки программного обеспечения Методы и средства получения дополнительной информации Общая методика отладки программного обеспечения
12.	Составление программной документации	Виды программных документов Пояснительная записка Руководство пользователя Руководство системного программиста Отчет по научно-исследовательской работе Основные правила оформления текстовых документов

6.2.2. Содержание практических занятий

№ п/п	Наименование темы (раздела) дисциплины	Содержание практического занятия
1.	Технология программирования (ТП). Основные понятия и подходы	<ol style="list-style-type: none"> 1. Простые и сложные программы 2. Основные этапы развития ТП 3. Проблемы разработки сложных программных систем 4. Блочный-иерархический подход к созданию сложных систем 5. Жизненный цикл (ЖЦ) и этапы разработки программного обеспечения

		<ul style="list-style-type: none"> 6. Модели ЖЦ, их особенности, плюсы и минусы 7. Ускорение разработки программного обеспечения. 8. Технология RAD 9. Оценка качества процессов создания программного обеспечения
2.	Приёмы обеспечения технологичности программных продуктов	<ul style="list-style-type: none"> 1. Понятие технологичности программного обеспечения 2. Модули и их свойства 3. Нисходящая и восходящая разработка программного обеспечения 4. Структурное и «неструктурное» программирование. 5. Средства описания структурных алгоритмов 6. Стилль оформления программы 7. Эффективность и технологичность Программирование «с защитой от ошибок»
3.	Определение требований к программному обеспечению и исходных данных для его проектирования	<ul style="list-style-type: none"> 1. Классификация программных продуктов по функциональному признаку 2. Основные эксплуатационные требования к программным продуктам 3. Предпроектные исследования предметной области 4. Разработка технического задания 5. Принципиальные решения начальных этапов проектирования
4.	Анализ требований и определение спецификаций программного обеспечения при структурном подходе	<ul style="list-style-type: none"> 1. Функциональные диаграммы 2. Диаграммы потоков данных 3. Структуры данных и диаграммы отношений компонентов данных 4. Математические модели задач, разработка или выбор методов решения
5.	Проектирование программного обеспечения при структурном подходе	<ul style="list-style-type: none"> 1. Диаграммы переходов состояний 2. Функциональные диаграммы 3. Диаграммы потоков данных 4. Структуры данных и диаграммы отношений компонентов данных 5. Математические модели задач, разработка или выбор методов решения
6.	Анализ требований и определение спецификаций программного обеспечения при объектном подходе	<ul style="list-style-type: none"> 1. UML стандартный язык описания разработки программных продуктов с использованием объектного подхода 2. Определение «вариантов использования» 3. Построение концептуальной модели предметной области 4. Описание поведения. 5. Системные события и операции
7.	Проектирование	<ul style="list-style-type: none"> 1. Определение отношений между объектами

	программного обеспечения при объектном подходе	<ol style="list-style-type: none"> 2. Уточнение отношений классов Проектирование классов 3. Компоновка программных компонентов 4. Проектирование размещения программных компонентов для распределенных программных систем 5. Особенность спиральной модели разработки. 6. Реорганизация проекта
8.	Разработка пользовательских интерфейсов	<ol style="list-style-type: none"> 1. Психофизические особенности человека, связанные с восприятием, запоминанием и обработкой информации 2. Пользовательская и программная модели интерфейса 3. Классификации диалогов и общие принципы их разработки 4. Основные компоненты графических пользовательских интерфейсов 5. Реализация диалогов в графическом пользовательском интерфейсе 6. Пользовательские интерфейсы прямого манипулирования и их проектирование 7. Интеллектуальные элементы пользовательских интерфейсов
9.	Пример разработки приложения Windows «Записная книжка»	<ol style="list-style-type: none"> 1. Разработка технического задания 2. Анализ предметной области, уточнение спецификаций и разработка структурной схемы 3. Проектирование интерфейса пользователя 4. Проектирование классов приложения
10.	Тестирование программных продуктов	<ol style="list-style-type: none"> 1. Виды контроля качества разрабатываемого программного обеспечения 2. Ручной контроль программного обеспечения 3. Структурное тестирование 4. Функциональное тестирование Тестирование модулей и комплексное тестирование 5. Оценочное тестирование
11.	Отладка программного обеспечения	<ol style="list-style-type: none"> 1. Отладка программного обеспечения 2. Классификация ошибок 3. Методы отладки программного обеспечения 4. Методы и средства получения дополнительной информации 5. Общая методика отладки программного обеспечения
12.	Составление программной документации	<ol style="list-style-type: none"> 1. Виды программных документов 2. Пояснительная записка 3. Руководство пользователя 4. Руководство системного программиста 5. Отчет по научно-исследовательской работе

		6. Основные правила оформления текстовых документов
--	--	---

6.2.3. Содержание самостоятельной работы

№ п/п	Наименование темы (раздела) дисциплины	Формы и тематика самостоятельной работы
1.	Технология программирования (ТП). Основные понятия и подходы	Технологии и методологии программирования Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
2.	Приёмы обеспечения технологичности программных продуктов	Структурное и «неструктурное» программирование. Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
3.	Определение требований к программному обеспечению и исходных данных для его проектирования	Разработка технического задания Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
4.	Анализ требований и определение спецификаций программного обеспечения при структурном подходе	Математические модели задач Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
5.	Проектирование программного обеспечения при структурном подходе	Структуры данных Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
6.	Анализ требований и определение спецификаций программного обеспечения при объектном подходе	Построение концептуальной модели предметной области Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
7.	Проектирование программного обеспечения при объектном подходе	Особенность спиральной модели разработки. Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
8.	Разработка пользовательских интерфейсов	Интеллектуальные элементы пользовательских интерфейсов Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
9.	Пример разработки	Проектирование интерфейса пользователя

	приложения Windows «Записная книжка»	Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами Индивидуальные задания
10.	Тестирование программных продуктов	Структурное тестирование Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
11.	Отладка программного обеспечения	Методы отладки программного обеспечения Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами
12.	Составление программной документации	Руководство пользователя Реферирование литературы Работа со справочными материалами Работа с Интернет-ресурсами Индивидуальные задания

7. Фонд оценочных средств для проведения текущей и промежуточной аттестации обучающихся по дисциплине (модулю)

Предусмотрены следующие виды контроля качества освоения конкретной дисциплины:

- текущий контроль успеваемости
- промежуточная аттестация обучающихся по дисциплине

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине оформлен в приложении к рабочей программе дисциплины

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины в процессе обучения.

7.1. Паспорт фонда оценочных средств для проведения текущей аттестации по дисциплине (модулю)

№ п/п	Контролируемые разделы (темы)	Наименование оценочного средства
1.	Технология программирования (ТП). Основные понятия и подходы	Вопросы к практическому занятию, интерактивное занятие
2.	Приёмы обеспечения технологичности программных продуктов	Вопросы к практическому занятию, практические задания, текущее тестирование.
3.	Определение требований к программному обеспечению и исходных данных для его проектирования	Вопросы к практическому занятию, практические задания, текущее тестирование.
4.	Анализ требований и определение спецификаций	Вопросы к практическому занятию, практические задания, текущее тестирование.

	программного обеспечения при структурном подходе	
5.	Проектирование программного обеспечения при структурном подходе	Вопросы к практическому занятию, практические задания, текущее тестирование.
6.	Анализ требований и определение спецификаций программного обеспечения при объектном подходе	Вопросы к практическому занятию, практические задания, текущее тестирование.
7.	Проектирование программного обеспечения при объектном подходе	Вопросы к практическому занятию, интерактивные занятия, текущее тестирование.
8.	Разработка пользовательских интерфейсов	Вопросы к практическому занятию, текущее тестирование, информационный проект
9.	Пример разработки приложения Windows «Записная книжка»	Вопросы к практическому занятию, контрольная работа, текущее тестирование.
10.	Тестирование программных продуктов	Вопросы к практическому занятию, интерактивные занятия, текущее тестирование.
11.	Отладка программного обеспечения	Вопросы к практическому занятию, практические задания, текущее тестирование.
12.	Составление программной документации	Вопросы к практическому занятию, практические задания, текущее тестирование.

7.2. Контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности в процессе промежуточного контроля

Тема 1. Технология программирования (ТП). Основные понятия и подходы

Вопросы к практическим занятиям:

1. Простые и сложные программы
2. Основные этапы развития ТП
3. Проблемы разработки сложных программных систем
4. Блочный-иерархический подход к созданию сложных систем
5. Жизненный цикл (ЖЦ) и этапы разработки программного обеспечения
6. Модели ЖЦ, их особенности, плюсы и минусы
7. Ускорение разработки программного обеспечения.
8. Технология RAD
9. Оценка качества процессов создания программного обеспечения

Тема 2. Приёмы обеспечения технологичности программных продуктов

Вопросы к практическим занятиям:

1. Понятие технологичности программного обеспечения

2. Модули и их свойства
3. Нисходящая и восходящая разработка программного обеспечения
4. Структурное и «неструктурное» программирование.
5. Средства описания структурных алгоритмов
6. Стил оформлени программы

Тема 3. Определение требований к программному обеспечению и исходных данных для его проектирования

Вопросы к практическим занятиям:

1. Классификация программных продуктов по функциональному признаку
2. Основные эксплуатационные требования к программным продуктам
3. Предпроектные исследования предметной области
4. Разработка технического задания
5. Принципиальные решения начальных этапов проектирования

Тема 4. Анализ требований и определение спецификаций программного обеспечения при структурном подходе

Вопросы к практическим занятиям:

1. Функциональные диаграммы
2. Диаграммы потоков данных
3. Структуры данных и диаграммы отношений компонентов данных
4. Математические модели задач, разработка или выбор методов решения

Тема 5. Проектирование программного обеспечения при структурном подходе

Вопросы к практическим занятиям:

1. Диаграммы переходов состояний
2. Функциональные диаграммы
3. Диаграммы потоков данных
4. Структуры данных и диаграммы отношений компонентов данных
5. Математические модели задач, разработка или выбор методов решения

Тема 6. Анализ требований и определение спецификаций программного обеспечения при объектном подходе

Вопросы к практическим занятиям:

1. UML стандартный язык описания разработки программных продуктов с использованием объектного подхода
2. Определение «вариантов использования»
3. Построение концептуальной модели предметной области
4. Описание поведения.
5. Системные события и операции

Тема 7. Проектирование программного обеспечения при объектном подходе

Вопросы к практическим занятиям:

1. Определение отношений между объектами
2. Уточнение отношений классов Проектирование классов
3. Компоновка программных компонентов

4. Проектирование размещения программных компонентов для распределенных программных систем
5. Особенность спиральной модели разработки.
6. Реорганизация проекта

Тема 8. Разработка пользовательских интерфейсов

Вопросы к практическим занятиям:

1. Психологические особенности человека, связанные с восприятием, запоминанием и обработкой информации
2. Пользовательская и программная модели интерфейса
3. Классификации диалогов и общие принципы их разработки
4. Основные компоненты графических пользовательских интерфейсов
5. Реализация диалогов в графическом пользовательском интерфейсе
6. Пользовательские интерфейсы прямого манипулирования и их проектирование
7. Интеллектуальные элементы пользовательских интерфейсов

Примерная тематика презентаций (информационных проектов)

1. Алгоритмы: определение, свойства.
2. Жизненный цикл программного обеспечения.
3. Обзор языков программирования.
4. Способы описания алгоритмов.
5. Структура программы на языке Си.
6. Типы данных в языке Си. Определение простых переменных в программе.
7. Классы памяти объектов языка Си.
8. Арифметические операции языка Си.
9. Логические операции языка Си
10. Операции языка Си: приведения типа, условная, перечисления, взятия размера.
11. Операторы языка Си: условный, оператор-переключатель.
12. Операторы циклов в языке Си.
13. Одномерные массивы в Си: определение, способы инициализации.
14. Типовые операции с одномерными массивами: поиск минимума, сумма элементов
15. Двумерные массивы в языке Си: определение, способы инициализации.
16. Указатели: назначение, определение, инициализация.
17. Динамическое выделение памяти.
18. Функции ввода-вывода в C++.
19. Универсальный метод выделения динамической памяти под двумерные массивы.
20. Строки в языке Си: определение, инициализация, кодирование символов.

Тема 9. Пример разработки приложения Windows «Записная книжка»

Вопросы к практическим занятиям:

1. Разработка технического задания
2. Анализ предметной области, уточнение спецификаций и разработка структурной схемы
3. Проектирование интерфейса пользователя
4. Проектирование классов приложения

Тематика контрольных работ

1. Программа: Решение алгебраических уравнений 1 -й, 2-й, 3-й и 4-й степени.
2. Программа: Решение системы N линейных алгебраических уравнений с N неизвестными.
3. Программа: Операции с алгебраическими матрицами.
4. Программа: Двоичный калькулятор.
5. Программа: Восьмеричный калькулятор.
6. Программа: Шестнадцатеричный калькулятор.
7. Программа: Электронный кассовый аппарат.
8. Программа: Переводы единиц измерения.
9. Программа: Оболочка для тестирования
10. Программа: Оболочка для анкетирования.
11. Программа: Календарь с часами.
12. Программа: Многофункциональные электронные часы.
13. Программа: Многофункциональный секундомер.
14. Программа: Протокол соревнований.
15. Программа: Таблица чемпионата.
16. Программа: Поиск сбойных участков на диске.
17. Форма, ее свойства и примеры использования.
18. Возможности и примеры использования окон редактирования иметок.
19. Возможности и примеры использования кнопок, флажков, переключателей.
20. Возможности и примеры использования списков.

Тема 10. Тестирование программных продуктов

Вопросы к практическим занятиям:

1. Виды контроля качества разрабатываемого программного обеспечения
2. Ручной контроль программного обеспечения
3. Структурное тестирование
4. Функциональное тестирование Тестирование модулей и комплексное тестирование
5. Оценочное тестирование

Тема 11. Отладка программного обеспечения

Вопросы к практическим занятиям:

1. Отладка программного обеспечения
2. Классификация ошибок
3. Методы отладки программного обеспечения
4. Методы и средства получения дополнительной информации
5. Общая методика отладки программного обеспечения

Тема 12. Составление программной документации

Вопросы к практическим занятиям:

1. Виды программных документов
2. Пояснительная записка
3. Руководство пользователя
4. Руководство системного программиста
5. Отчет по научно-исследовательской работе
6. Основные правила оформления текстовых документов

Практические задания

1. Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.
2. Составить программу, печатающую значение *true*, если указанное высказывание является истинным, и *false* в противном случае:
сумма двух первых цифр заданного четырехзначного числа равна сумме двух его последних цифр.
3. Составить программу, которая печатает *true*, если точка с координатами (x, y) принадлежит заштрихованной области, и *false* в противном случае:
4. Написать программу, которая бы по введенному номеру единицы измерения (1 — килограмм, 2 — миллиграмм, 3 — грамм, 4 — тонна, 5 — центнер) и массе М выдавала бы соответствующее значение массы в килограммах.
5. Даны два действительных числа. Если они равны, то занулить, в противном случае наименьшему числу присвоить значение наибольшего числа.
6. Дана последовательность действительных чисел a_1, a_2, \dots, a_n . Указать те элементы, которые принадлежат отрезку $[c, d]$.
7. Дан массив целых чисел из 12 элементов. Упорядочить массив по убыванию (метод Шелла).
8. Квадратная матрица, симметричная относительно главной диагонали, задана верхним треугольником в виде одномерного массива. Восстановить исходную матрицу и напечатать результат.
9. Дан текст. Подсчитать, сколько раз среди символов текста встречается символ «пробел».
10. В заданном тексте удалить часть текста, заключенную в скобки, вместе со скобками.

Текущее тестирование

1. Целью программирования является
 - а) представление фактов и идей;
 - б) процесс обработки данных;
 - в) последовательность состояний;
 - г) фиксирование документов.
2. Способность безотказно выполнять определенные функции при заданных условиях — это
 - а) надежность;
 - б) безошибочность;
 - в) дефект;
 - г) правильность.
3. Технология разработки программных средств — это
 - а) программная инженерия;
 - б) технологический процесс;
 - в) компьютерная программа;
 - г) технология программирования.
4. Широкое распространение технологии программирования получили
 - а) в 50-е годы;
 - б) в 60-е годы;
 - в) в 70-е годы;
 - г) в 80-е годы.

5. Какой интеллектуальной возможности человека для разработки ПС не существует?
- а) способность к дедукции;
 - б) способность к индукции;
 - в) способность к перебору;
 - г) способность к абстракции.
6. Что такое система?
- а) бесконечные последовательности, объединенные в один элемент;
 - б) совокупность взаимодействующих элементов;
 - в) пути взаимодействия между элементами;
 - г) обобщенные требования утверждений.
7. Причина ошибок в программных средствах:
- а) большое число элементов в системе;
 - б) сбой в аппаратном средстве;
 - в) большое количество разработчиков;
 - г) неправильный перевод информации.
8. Какого пути борьбы с ошибками не существует?
- а) замены аппаратного обеспечения;
 - б) сужения пространства перебора;
 - в) обеспечения уровня подготовки разработчика;
 - г) контроля правильности перевода.
9. Жизненный цикл программного средства — это
- а) время продажи ПС пользователям;
 - б) период эксплуатации ПС пользователем на компьютере;
 - в) период разработки и эксплуатации ПС от замысла до полного прекращения;
 - г) общее время разработки ПС от замысла до готового продукта.
10. Какого подхода к организации процесса создания и использования ПС не существует?
- а) водопадного;
 - б) исследовательского;
 - в) сборочного;
 - г) модульного.
11. Как называется экземпляр разработанного ПС?
- а) программный продукт;
 - б) программное изделие;
 - в) готовый проект;
 - г) прикладное приложение.
12. Что не относится к критериям качества ПС?
- а) функциональность;
 - б) надежность;
 - в) стоимость;
 - г) мобильность.
13. Какой задачи для предупреждения ошибок не существует?
- а) изменение программного кода;

- б) борьба со сложностью;
- в) обеспечение точности перевода;
- г) обеспечение контроля принимаемых решений.

14. Какой контроль принимаемых решений существует при разработке ПС?

- а) текущий;
- б) внешний;
- в) внутренний;
- г) смежный.

15. Результат процесса формулирования требований к ПС называется

- а) технологическим процессом;
- б) внешним описанием;
- в) файлом помощи;
- г) внутренним описанием.

16. Какого способа разработки определения требований к ПС не существует?

- а) управляемая разработка пользователем и разработчиком в равной степени;
- б) управляемая пользователем разработка;
- в) контролируемая пользователем разработка;
- г) независимая от пользователя разработка.

17. Стандартизированный набор простых свойств определения качества ПС называется:

- а) функционалами;
- б) примитивами;
- в) критериями;
- г) документами.

18. Каковы подкритерии сопровождаемости?

- а) мобильность и завершенность;
- б) точность и автономность;
- в) изучаемость и модифицируемость;
- г) устойчивость и эффективность.

19. Свойство, характеризующее программы ПС с точки зрения организации взаимосвязанных частей в единое целое — это

- а) модульность;
- б) системность;
- в) удобочитаемость;
- г) структурированность.

20. Что не входит в функциональную спецификацию?

- а) описание внешней информационной среды;
- б) определение внешних функций;
- в) описание исключительных ситуаций;
- г) описание критериев качества.

21. Какого метода контроля внешнего описания ПС не существует?

- а) статический просмотр;
- б) смежный;
- в) внутренний;

г) пользовательский.

22. Какого подхода к спецификации семантики функций не существует?

- а) табличного;
- б) геометрического;
- в) алгебраического;
- г) логического.

23. Что определяет нижняя часть таблицы решений?

- а) комбинацию выполняемых действий;
- б) комбинацию значений ситуаций;
- в) комбинацию произвольных значений;
- г) комбинацию состояний условий.

24. Какого вида семантики не существует?

- а) операционная;
- б) денотационная;
- в) формализованная;
- г) аксиоматическая.

25. Переписывание выражения с заменой одного вхождения на другое называется

- а) композицией;
- б) нотацией;
- в) интерпретацией;
- г) подстановкой.

Типовые задания к интерактивным занятиям

1. Основы объектно-ориентированного программирования
2. Методы объектно-ориентированного программирования
3. Синтаксические закономерности проектирования
4. Создание приложений для Windows в Delphi
5. Среда разработки Delphi
6. Создание приложений в Delphi
7. Компоненты графического интерфейса

7.3 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Все задания, используемые для текущего контроля формирования компетенций условно можно разделить на две группы:

1. задания, которые в силу своих особенностей могут быть реализованы только в процессе обучения на занятиях (например, дискуссия, круглый стол, диспут, мини-конференция);
2. задания, которые дополняют теоретические вопросы (практические задания, проблемно-аналитические задания, тест).

Выполнение всех заданий является необходимым для формирования и контроля знаний, умений и навыков. Поэтому, в случае невыполнения заданий в процессе обучения, их необходимо «отработать» до зачета (экзамена). Вид заданий, которые необходимо выполнить для ликвидации «задолженности» определяется в индивидуальном порядке, с учетом причин невыполнения.

1. Требование к теоретическому устному ответу

Оценка знаний предполагает дифференцированный подход к студенту, учет его индивидуальных способностей, степень усвоения и систематизации основных понятий и категорий по дисциплине. Кроме того, оценивается не только глубина знаний поставленных вопросов, но и умение использовать в ответе практический материал. Оценивается культура речи, владение навыками ораторского искусства.

Критерии оценивания: последовательность, полнота, логичность изложения, анализ различных точек зрения, самостоятельное обобщение материала, использование профессиональных терминов, культура речи, навыки ораторского искусства. Изложение материала без фактических ошибок.

Оценка «отлично» ставится в случае, когда материал излагается исчерпывающе, последовательно, грамотно и логически стройно, при этом раскрываются не только основные понятия, но и анализируются точки зрения различных авторов. Обучающийся не затрудняется с ответом, соблюдает культуру речи.

Оценка «хорошо» ставится, если обучающийся твердо знает материал, грамотно и по существу излагает его, знает практическую базу, но при ответе на вопрос допускает несущественные погрешности.

Оценка «удовлетворительно» ставится, если обучающийся освоил только основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении материала, затрудняется с ответами, показывает отсутствие должной связи между анализом, аргументацией и выводами.

Оценка «неудовлетворительно» ставится, если обучающийся не отвечает на поставленные вопросы.

2. Творческие задания

Эссе – это небольшая по объему письменная работа, сочетающая свободные, субъективные рассуждения по определенной теме с элементами научного анализа. Текст должен быть легко читаем, но необходимо избегать нарочито разговорного стиля, сленга, шаблонных фраз. Объем эссе составляет примерно 2 – 2,5 стр. 12 шрифтом с одинарным интервалом (без учета титульного листа).

Критерии оценивания - оценка учитывает соблюдение жанровой специфики эссе, наличие логической структуры построения текста, наличие авторской позиции, ее научность и связь с современным пониманием вопроса, адекватность аргументов, стиль изложения, оформление работы. Следует помнить, что прямое заимствование (без оформления цитат) текста из Интернета или электронной библиотеки недопустимо.

Оценка «отлично» ставится в случае, когда определяется: наличие логической структуры построения текста (вступление с постановкой проблемы; основная часть, разделенная по основным идеям; заключение с выводами, полученными в результате рассуждения); наличие четко определенной личной позиции по теме эссе; адекватность аргументов при обосновании личной позиции, стиль изложения.

Оценка «хорошо» ставится, когда в целом определяется: наличие логической структуры построения текста (вступление с постановкой проблемы; основная часть, разделенная по основным идеям; заключение с выводами, полученными в результате рассуждения); но не прослеживается наличие четко определенной личной позиции по теме эссе; не достаточно аргументов при обосновании личной позиции.

Оценка «удовлетворительно» ставится, когда в целом определяется: наличие логической структуры построения текста (вступление с постановкой проблемы; основная часть, разделенная по основным идеям; заключение). Но не прослеживаются четкие выводы, нарушается стиль изложения.

Оценка «неудовлетворительно» ставится, если не выполнены никакие требования.

3. Требование к решению ситуационной, проблемной задачи (кейс-измерители)

Студент должен уметь выделить основные положения из текста задачи, которые

требуют анализа и служат условиями решения. Исходя из поставленного вопроса в задаче, попытаться максимально точно определить проблему и соответственно решить ее.

Задачи должны решаться студентами письменно. При решении задач также важно правильно сформулировать и записать вопросы, начиная с более общих и, кончая частными.

Критерии оценивания – оценка учитывает методы и средства, использованные при решении ситуационной, проблемной задачи.

Оценка «*отлично*» ставится в случае, когда обучающийся выполнил задание (решил задачу), используя в полном объеме теоретические знания и практические навыки, полученные в процессе обучения.

Оценка «*хорошо*» ставится, если обучающийся в целом выполнил все требования, но не совсем четко определяется опора на теоретические положения, изложенные в научной литературе по данному вопросу.

Оценка «*удовлетворительно*» ставится, если обучающийся показал положительные результаты в процессе решения задачи.

Оценка «*неудовлетворительно*» ставится, если обучающийся не выполнил все требования.

4. Интерактивные задания

Механизм проведения диспут-игры (ролевой (деловой) игры).

Необходимо разбиться на несколько команд, которые должны поочередно высказать свое мнение по каждому из заданных вопросов. Мнение высказывающейся команды засчитывается, если противоположная команда не опровергнет его контраргументами. Команда, чье мнение засчитано как верное (не получило убедительных контраргументов от противоположных команд), получает один балл. Команда, опровергнувшая мнение противоположной команды своими контраргументами, также получает один балл. Побеждает команда, получившая максимальное количество баллов.

Ролевая игра как правило имеет фабулу (ситуацию, казус), распределяются роли, подготовка осуществляется за 2-3 недели до проведения игры.

Критерии оценивания – оцениваются действия всех участников группы. Понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Соответствие реальной действительности решений, выработанных в ходе игры. Владение терминологией, демонстрация владения учебным материалом по теме игры, владение методами аргументации, умение работать в группе (умение слушать, конструктивно вести беседу, убеждать, управлять временем, бесконфликтно общаться), достижение игровых целей, (соответствие роли – при ролевой игре). Ясность и стиль изложения.

Оценка «*отлично*» ставится в случае, выполнения всех критериев.

Оценка «*хорошо*» ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Решения, выработанные в ходе игры, полностью соответствуют реальной действительности. Но некоторые объяснения не совсем аргументированы, нарушены нормы общения, нарушены временные рамки, нарушен стиль изложения.

Оценка «*удовлетворительно*» ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия в целом соответствуют заданным целям. Однако, решения, выработанные в ходе игры, не совсем соответствуют реальной действительности. Некоторые объяснения не совсем аргументированы, нарушены временные рамки, нарушен стиль изложения.

Оценка «*неудовлетворительно*» ставится, если обучающиеся не понимают проблему, их высказывания не соответствуют заданным целям.

5. Комплексное проблемно-аналитическое задание

Задание носит проблемно-аналитический характер и выполняется в три этапа. На первом из них необходимо ознакомиться со специальной литературой.

Целесообразно также повторить учебные материалы лекций и семинарских занятий

по темам, в рамках которых предлагается выполнение данного задания.

На втором этапе выполнения работы необходимо сформулировать проблему и изложить авторскую версию ее решения, на основе полученной на первом этапе информации.

Третий этап работы заключается в формулировке собственной точки зрения по проблеме. Результат третьего этапа оформляется в виде аналитической записки (объем: 2-2,5 стр.; 14 шрифт, 1,5 интервал).

Критерий оценивания - оценка учитывает: понимание проблемы, уровень раскрытия поставленной проблемы в плоскости теории изучаемой дисциплины, умение формулировать и аргументировано представлять собственную точку зрения, выполнение всех этапов работы.

Оценка «отлично» ставится в случае, когда обучающийся демонстрирует полное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «хорошо» ставится, если обучающийся демонстрирует значительное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «удовлетворительно» ставится, если обучающийся, демонстрирует частичное понимание проблемы, большинство требований, предъявляемых к заданию, выполнены

Оценка «неудовлетворительно» ставится, если обучающийся демонстрирует непонимание проблемы, многие требования, предъявляемые к заданию, не выполнены.

6. Исследовательский проект

Исследовательский проект – проект, структура которого приближена к формату научного исследования и содержит доказательство актуальности избранной темы, определение научной проблемы, предмета и объекта исследования, целей и задач, методов, источников, историографии, обобщение результатов, выводы.

Результаты выполнения исследовательского проекта оформляется в виде реферата (объем: 12-15 страниц; 14 шрифт, 1,5 интервал).

Критерии оценивания - поскольку структура исследовательского проекта максимально приближена к формату научного исследования, то при выставлении учитывается доказательство актуальности темы исследования, определение научной проблемы, объекта и предмета исследования, целей и задач, источников, методов исследования, выдвижение гипотезы, обобщение результатов и формулирование выводов, обозначение перспектив дальнейшего исследования.

Оценка «отлично» ставится в случае, когда обучающийся демонстрирует полное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «хорошо» ставится, если обучающийся демонстрирует значительное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «удовлетворительно» ставится, если обучающийся, демонстрирует частичное понимание проблемы, большинство требований, предъявляемых к заданию, выполнены

Оценка «неудовлетворительно» ставится, если обучающийся демонстрирует непонимание проблемы, многие требования, предъявляемые к заданию, не выполнены.

7. Информационный проект (презентация):

Информационный проект – проект, направленный на стимулирование учебно-познавательной деятельности студента с выраженной эвристической направленностью (поиск, отбор и систематизация информации об объекте, оформление ее для презентации). Итоговым продуктом проекта может быть письменный реферат, электронный реферат с иллюстрациями, слайд-шоу, мини-фильм, презентация и т.д.

Информационный проект отличается от исследовательского проекта, поскольку представляет собой такую форму учебно-познавательной деятельности, которая отличается ярко выраженной эвристической направленностью.

Критерии оценивания - при выставлении оценки учитывается самостоятельный

поиск, отбор и систематизация информации, раскрытие вопроса (проблемы), ознакомление студенческой аудитории с этой информацией (представление информации), ее анализ и обобщение, оформление, полные ответы на вопросы аудитории с примерами.

Оценка «*отлично*» ставится в случае, когда обучающийся полностью раскрывает вопрос (проблему), представляет информацию систематизировано, последовательно, логично, взаимосвязано, использует более 5 профессиональных терминов, широко использует информационные технологии, ошибки в информации отсутствуют, дает полные ответы на вопросы аудитории с примерами.

Оценка «*хорошо*» ставится, если обучающийся раскрывает вопрос (проблему), представляет информацию систематизировано, последовательно, логично, взаимосвязано, использует более 2 профессиональных терминов, достаточно использует информационные технологии, допускает не более 2 ошибок в изложении материала, дает полные или частично полные ответы на вопросы аудитории.

Оценка «*удовлетворительно*» ставится, если обучающийся, раскрывает вопрос (проблему) не полностью, представляет информацию не систематизировано и не совсем последовательно, использует 1-2 профессиональных термина, использует информационные технологии, допускает 3-4 ошибки в изложении материала, отвечает только на элементарные вопросы аудитории без пояснений.

Оценка «*неудовлетворительно*» ставится, если вопрос не раскрыт, представленная информация логически не связана, не используются профессиональные термины, допускает более 4 ошибок в изложении материала, не отвечает на вопросы аудитории.

8. Дискуссионные процедуры

Круглый стол, дискуссия, полемика, диспут, дебаты, мини-конференции являются средствами, позволяющими включить обучающихся в процесс обсуждения спорного вопроса, проблемы и оценить их умение аргументировать собственную точку зрения. Задание дается заранее, определяется круг вопросов для обсуждения, группы участников этого обсуждения.

Дискуссионные процедуры могут быть использованы для того, чтобы студенты:

– лучше поняли усвояемый материал на фоне разнообразных позиций и мнений, не обязательно достигая общего мнения;

– смогли постичь смысл изучаемого материала, который иногда чувствуют интуитивно, но не могут высказать вербально, четко и ясно, или конструировать новый смысл, новую позицию;

– смогли согласовать свою позицию или действия относительно обсуждаемой проблемы.

Критерии оценивания – оцениваются действия всех участников группы. Понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Соответствие реальной действительности решений, выработанных в ходе игры. Владение терминологией, демонстрация владения учебным материалом по теме игры, владение методами аргументации, умение работать в группе (умение слушать, конструктивно вести беседу, убеждать, управлять временем, бесконфликтно общаться), достижение игровых целей, (соответствие роли – при ролевой игре). Ясность и стиль изложения.

Оценка «*отлично*» ставится в случае, когда все требования выполнены в полном объеме.

Оценка «*хорошо*» ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Решения, выработанные в ходе игры, полностью соответствуют реальной действительности. Но некоторые объяснения не совсем аргументированы, нарушены нормы общения, нарушены временные рамки, нарушен стиль изложения.

Оценка «*удовлетворительно*» ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия в целом соответствуют заданным целям. Однако, решения, выработанные в ходе игры, не совсем соответствуют реальной

действительности. Некоторые объяснения не совсем аргументированы, нарушены временные рамки, нарушен стиль изложения.

Оценка «*неудовлетворительно*» ставится, если обучающиеся не понимают проблему, их высказывания не соответствуют заданным целям.

9. Тестирование

Является одним из средств контроля знаний, обучающихся по дисциплине.

Критерии оценивания – правильный ответ на вопрос.

Оценка «*отлично*» ставится в случае, если правильно выполнено 90-100% заданий.

Оценка «*хорошо*» ставится, если правильно выполнено 70-89% заданий.

Оценка «*удовлетворительно*» ставится в случае, если правильно выполнено 50-69% заданий.

Оценка «*неудовлетворительно*» ставится, если правильно выполнено менее 50% заданий.

10. Требование к письменному опросу (контрольной работе)

Оценивается не только глубина знаний поставленных вопросов, но и умение изложить письменно.

Критерии оценивания: последовательность, полнота, логичность изложения, анализ различных точек зрения, самостоятельное обобщение материала. Изложение материала без фактических ошибок.

Оценка «*отлично*» ставится в случае, когда соблюдены все критерии.

Оценка «*хорошо*» ставится, если обучающийся твердо знает материал, грамотно и по существу излагает его, знает практическую базу, но допускает несущественные погрешности.

Оценка «*удовлетворительно*» ставится, если обучающийся освоил только основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении материала, затрудняется с ответами, показывает отсутствие должной связи между анализом, аргументацией и выводами.

Оценка «*неудовлетворительно*» ставится, если обучающийся не отвечает на поставленные вопросы.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

8.1. Основная учебная литература:

1. Ковалевская, Е. В. Методы программирования : учебное пособие / Е. В. Ковалевская, Н. В. Комлева. — Москва : Евразийский открытый институт, 2011. — 320 с. — ISBN 978-5-374-00356-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/10784.html>

2. Золин, А. Г. Языки и методы программирования. Введение в разработку на C++ (первый семестр) : учебное пособие / А. Г. Золин, А. Е. Колоденкова, Е. А. Халикова. — Самара : Самарский государственный технический университет, ЭБС АСВ, 2020. — 128 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/105256.html>

3. Киселева, Т. В. Программная инженерия. Часть 1 : учебное пособие / Т. В. Киселева. — Ставрополь : Северо-Кавказский федеральный университет, 2017. — 137 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/69425.html>

4. Соловьев, Н. А. Введение в программную инженерию : учебное пособие / Н. А. Соловьев, Л. А. Юркевская. — Оренбург : Оренбургский государственный университет, ЭБС АСВ, 2017. — 112 с. — ISBN 978-5-7410-1685-5. — Текст : электронный // Электронно-

библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/71267.html>

8.2. Дополнительная учебная литература:

1. Полетайкин, А. Н. Учебно-методическое пособие по выполнению лабораторных работ по дисциплине «Программная инженерия». Часть I. Реализация жизненного цикла программного обеспечения : учебно-методическое пособие / А. Н. Полетайкин. — Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2016. — 97 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/69565.html>

2. Русанова Я.М. С++ как второй язык в обучении приемам и технологиям программирования [Электронный ресурс]/ Русанова Я.М., Чердынцева М.И.— Электрон. текстовые данные.— Ростов-на-Дону: Южный федеральный университет, 2010.— 200 с.— Режим доступа: <http://www.iprbookshop.ru/47120> .

3. Кудинов Ю.И. Основы алгоритмизации и программирования. Часть 1 [Электронный ресурс]: учебное пособие/ Кудинов Ю.И., Келина А.Ю.— Электрон. текстовые данные.— Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2013.— 71 с.— Режим доступа: <http://www.iprbookshop.ru/55121> .

4. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 285 с. — ISBN 978-5-4486-0513-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/79706.html>

8.3. Периодические издания

1. Журнал «Компьютерра» <http://www.computerra.ru>

9. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), необходимых для освоения дисциплины (модуля)

1. Федеральный портал «Российское образование». <http://www.edu.ru/>
2. Электронно-библиотечная система IPR BOOKS <https://www.iprbookshop.ru/>

10. Методические указания для обучающихся по освоению дисциплины (модуля)

Успешное освоение данного курса базируется на рациональном сочетании нескольких видов учебной деятельности – лекционных занятий, практических занятий, самостоятельной работы. При этом самостоятельную работу следует рассматривать одним из главных звеньев полноценного высшего образования, на которую отводится значительная часть учебного времени.

Самостоятельная работа студентов складывается из следующих составляющих:

- работа с основной и дополнительной литературой, с материалами интернета и конспектами лекций;
- внеаудиторная подготовка к контрольным работам, выполнение докладов, рефератов и курсовых работ;
- выполнение самостоятельных практических работ;
- подготовка к экзаменам (зачетам) непосредственно перед ними.

Для правильной организации работы необходимо учитывать порядок изучения разделов курса, находящихся в строгой логической последовательности. Поэтому хорошее усвоение одной части дисциплины является предпосылкой для успешного перехода к следующей. Задания, проблемные вопросы, предложенные для изучения дисциплины, в том числе и для самостоятельного выполнения, носят междисциплинарный характер и базируются, прежде всего, на причинно-следственных связях между компонентами

окружающего нас мира. В течение семестра необходимо подготовить рефераты с использованием рекомендуемой основной и дополнительной литературы и сдать рефераты для проверки преподавателю. Важным составляющим в изучении данного курса является решение различных задач и работа над проблемно-аналитическими заданиями, что предполагает знание соответствующей научной терминологии.

При выполнении докладов, творческих, информационных, исследовательских проектов особое внимание следует обращать на подбор источников информации и методику работы с ними.

Для успешной сдачи экзамена (зачета) рекомендуется соблюдать следующие правила:

- Подготовка к экзамену (зачету) должна проводиться систематически, в течение всего семестра.

- Интенсивная подготовка должна начаться не позднее, чем за месяц до экзамена.

- Время непосредственно перед экзаменом лучше использовать таким образом, чтобы оставить последний день свободным для повторения курса в целом, для систематизации материала и доработки отдельных вопросов.

На экзамене (зачете) высокую оценку получают студенты, использующие данные, полученные в процессе выполнения самостоятельных работ, а также использующие собственные выводы на основе изученного материала.

Учитывая значительный объем теоретического материала, студентам рекомендуется регулярное посещение и подробное конспектирование лекций.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

1. Microsoft Windows Server;
2. Семейство ОС Microsoft Windows;
3. Libre Office свободно распространяемый офисный пакет с открытым исходным кодом;
4. Информационно-справочная система: Система КонсультантПлюс (КонсультантПлюс);
5. Информационно-правовое обеспечение Гарант: Электронный периодический справочник «Система ГАРАНТ» (Система ГАРАНТ);

Перечень используемого программного обеспечения указан в п.12 данной рабочей программы дисциплины.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

12.1. Учебная аудитория для проведения учебных занятий, предусмотренных образовательной программой, оснащенная оборудованием и техническими средствами обучения.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; комплект мебели для преподавателя; доска (маркерная).

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; наушники; телевизор.

Перечень лицензионного программного обеспечения, в том числе отечественного производства и свободно распространяемого программного обеспечения:

Windows Server 2016, Windows 10, Microsoft Office, КонсультантПлюс, Система ГАРАНТ, Kaspersky Endpoint Security, Microsoft Windows Server, Microsoft Project, Spider Project, EclipseIDEforJavaEEDevelopers, AndroidStudio, IntelliJIDEA, Adobe Acrobat Reader DC, Google Chrome, LibreOffice, Skype, Gimp, Paint.net, AnyLogic, Inkscape, Microsoft Visual Studio Community, Denver, GNU Octave, PostgreSQL, Ramus.

Подключение к сети «Интернет» и обеспечение доступа в электронную информационно-образовательную среду ММУ.

12.2. Помещение для самостоятельной работы обучающихся.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; комплект мебели для преподавателя; доска (маркерная).

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; колонки; проектор, экран.

Перечень лицензионного программного обеспечения, в том числе отечественного производства:

Windows Server 2016, Windows 10, Microsoft Office, КонсультантПлюс, Система ГАРАНТ, Kaspersky Endpoint Security.

Перечень свободно распространяемого программного обеспечения:

Adobe Acrobat Reader DC, Google Chrome, LibreOffice, Skype, Zoom, Gimp, Paint.net, AnyLogic, Inkscape.

Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ММУ.

13. Образовательные технологии, используемые при освоении дисциплины

Для освоения дисциплины используются как традиционные формы занятий – лекционные занятия (типы лекций – установочная, вводная, текущая, заключительная, обзорная; виды лекций – проблемная, визуальная, лекция конференция, лекция консультация) и практические занятия, так и активные и интерактивные формы занятий – диспуты, решение ситуационных задач, ролевые игры и разбор конкретных ситуаций.

На учебных занятиях используются технические средства обучения – проектор, ноутбук, проекционный экран, колонки для демонстрации слайдов, видеосюжетов и др. Тестирование обучаемых может осуществляться с использованием компьютерного оборудования университета.

13.1. В освоении учебной дисциплины используются следующие традиционные образовательные технологии:

- чтение проблемно-информационных лекций с использованием доски и видеоматериалов;
- практические занятия;
- контрольные опросы;
- консультации;
- самостоятельная работа с учебной литературой;
- подготовка и обсуждение рефератов, презентаций;
- тестирование по основным темам дисциплины.

13.2. Активные и интерактивные методы и формы обучения

Из перечня видов: («мозговой штурм», анализ НПА, анализ проблемных ситуаций, анализ конкретных ситуаций, инциденты, имитация коллективной профессиональной деятельности, разыгрывание ролей, творческая работа, связанная с освоением дисциплины, ролевая игра, круглый стол, диспут, беседа, дискуссия, мини-конференция и др.) используются следующие:

- анализ проблемных-аналитических заданий,
- творческие задания;
- дискуссия.

13.3. Особенности обучения инвалидов и лиц с ограниченными возможностями

здоровья (ОВЗ)

При организации обучения по дисциплине учитываются особенности организации взаимодействия с инвалидами и лицами с ограниченными возможностями здоровья (далее – инвалиды и лица с ОВЗ) с целью обеспечения их прав. При обучении учитываются особенности их психофизического развития, индивидуальные возможности и при необходимости обеспечивается коррекция нарушений развития и социальная адаптация указанных лиц.

Выбор методов обучения определяется содержанием обучения, уровнем методического и материально-технического обеспечения, особенностями восприятия учебной информации студентов-инвалидов и студентов с ограниченными возможностями здоровья и т.д. В образовательном процессе используются социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение и дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

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

Автономная некоммерческая организация высшего образования
«МОСКОВСКИЙ МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ»

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ**

Технологии и методы программирования

<i>Направление подготовки</i>	Бизнес-информатика
<i>Код</i>	38.03.05
<i>Направленность (профиль)</i>	Информационные системы и технологии в бизнесе
<i>Квалификация выпускника</i>	бакалавр

Москва
2024

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Общепрофессиональные		ОПК-3

2. Компетенции и индикаторы их достижения

Компетенция	Индикаторы достижения компетенций	Планируемые результаты обучения по дисциплине
ОПК-3	Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации	ОПК-3.1 Знает понятие, виды и особенности продуктов и услуг в сфере ИКТ; основы алгоритмизации, современные методологии разработки программных средств; этапы разработки программных средств; методы обеспечения информационной безопасности. ОПК-3.2 Умеет разрабатывать алгоритмы и программы для практической реализации продуктов и услуг в сфере ИКТ. ОПК-3.3 Владеет методами управления процессами создания и использования продуктов и услуг в сфере ИКТ, в частности, навыками разработки алгоритмов и программ для их практической реализации.

3. Описание планируемых результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки).

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ОПК-3		
	- об эволюции и тенденциях развития технологий разработки программного обеспечения; - о средствах автоматизации	- использовать критерии технологичности ПО, методы и средства ее повышения; - определять состав и порядок определения требований к	- навыками использования программных средств для решения практических задач; - навыками программирования на языке высокого уровня;

	<p>разработки программного обеспечения (CASE-технологии);</p> <ul style="list-style-type: none"> - об объектном подходе к спецификации, проектированию и тестированию программного обеспечения. - базовые алгоритмы обработки данных; - основные технологии и методы программирования. - этапы и модели жизненного цикла программного обеспечения (ПО); - основные показатели качества ПО 	<p>программному обеспечению;</p> <ul style="list-style-type: none"> - самостоятельно выбирать необходимые технологии программирования для решения конкретных задач; - осуществлять организацию процессов тестирования и отладки ПО, методы структурного и функционального тестирования; - использовать состав и принципы составления программной документации. 	<ul style="list-style-type: none"> - навыками составления технического задания на разработку ПО; - практическими навыками применения технологических приемов разработки программного обеспечения.
--	--	---	---

3.2. Критерии оценки результатов обучения по дисциплине

Шкала оценивания	Индикаторы достижения	Показатели оценивания результатов обучения
ОТЛИЧНО/ЗАЧТЕНО	Знает:	<ul style="list-style-type: none"> - студент глубоко и всесторонне усвоил материал, уверенно, логично, последовательно и грамотно его излагает, опираясь на знания основной и дополнительной литературы, - на основе системных научных знаний делает квалифицированные выводы и обобщения, свободно оперирует категориями и понятиями.
	Умеет:	<ul style="list-style-type: none"> - студент умеет самостоятельно и правильно решать учебно-профессиональные задачи или задания, уверенно, логично, последовательно и аргументировано излагать свое решение, используя научные понятия, ссылаясь на нормативную базу.
	Владеет:	<ul style="list-style-type: none"> - студент владеет рациональными методами (с использованием рациональных методик) решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении продемонстрировал навыки - выделения главного, - связкой теоретических положений с требованиями руководящих документов, - изложения мыслей в логической последовательности, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.

ХОРОШО/ЗАЧТНО	Знает:	<ul style="list-style-type: none"> - студент твердо усвоил материал, достаточно грамотно его излагает, опираясь на знания основной и дополнительной литературы, - затрудняется в формулировании квалифицированных выводов и обобщений, оперирует категориями и понятиями, но не всегда правильно их верифицирует.
	Умеет:	<ul style="list-style-type: none"> - студент умеет самостоятельно и в основном правильно решать учебно-профессиональные задачи или задания, уверенно, логично, последовательно и аргументировано излагать свое решение, не в полной мере используя научные понятия и ссылки на нормативную базу.
	Владеет:	<ul style="list-style-type: none"> - студент в целом владеет рациональными методами решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении смог продемонстрировать достаточность, но не глубинность навыков - выделения главного, - изложения мыслей в логической последовательности. - связи теоретических положений с требованиями руководящих документов, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
УДОВЛЕТВИТЕЛЬНО/ЗАЧТНО	Знает:	<ul style="list-style-type: none"> - студент ориентируется в материале, однако затрудняется в его изложении; - показывает недостаточность знаний основной и дополнительной литературы; - слабо аргументирует научные положения; - практически не способен сформулировать выводы и обобщения; - частично владеет системой понятий.
	Умеет:	<ul style="list-style-type: none"> - студент в основном умеет решить учебно-профессиональную задачу или задание, но допускает ошибки, слабо аргументирует свое решение, недостаточно использует научные понятия и руководящие документы.
	Владеет:	<ul style="list-style-type: none"> - студент владеет некоторыми рациональными методами решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении продемонстрировал недостаточность навыков - выделения главного, - изложения мыслей в логической последовательности. - связи теоретических положений с требованиями руководящих документов, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
Компетенция не достигнута		

НЕУДОВЛЕТВОРИТЕЛЬНО/ НЕЗАЧТЕНО	Знает:	- студент не усвоил значительной части материала; - не может аргументировать научные положения; - не формулирует квалифицированных выводов и обобщений; - не владеет системой понятий.
	Умеет:	студент не показал умение решать учебно-профессиональную задачу или задание.
	Владеет:	не выполнены требования, предъявляемые к навыкам, оцениваемым “удовлетворительно”.

4. Типовые контрольные задания и/или иные материалы для проведения промежуточной аттестации, необходимые для оценки знаний, умений, навыков и/или опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Тесты

1. Целью программирования является
 - а) представление фактов и идей;
 - б) процесс обработки данных;
 - в) последовательность состояний;
 - г) фиксирование документов.

2. Способность безотказно выполнять определенные функции при заданных условиях — это
 - а) надежность;
 - б) безошибочность;
 - в) дефект;
 - г) правильность.

3. Технология разработки программных средств — это
 - а) программная инженерия;
 - б) технологический процесс;
 - в) компьютерная программа;
 - г) технология программирования.

4. Широкое распространение технологии программирования получили
 - а) в 50-е годы;
 - б) в 60-е годы;
 - в) в 70-е годы;
 - г) в 80-е годы.

5. Какой интеллектуальной возможности человека для разработки ПС не существует?
 - а) способность к дедукции;
 - б) способность к индукции;
 - в) способность к перебору;
 - г) способность к абстракции.

6. Что такое система?

- а) бесконечные последовательности, объединенные в один элемент;
 - б) совокупность взаимодействующих элементов;
 - в) пути взаимодействия между элементами;
 - г) обобщенные требования утверждений.
7. Причина ошибок в программных средствах:
- а) большое число элементов в системе;
 - б) сбой в аппаратном средстве;
 - в) большое количество разработчиков;
 - г) неправильный перевод информации.
8. Какого пути борьбы с ошибками не существует?
- а) замены аппаратного обеспечения;
 - б) сужения пространства перебора;
 - в) обеспечения уровня подготовки разработчика;
 - г) контроля правильности перевода.
9. Жизненный цикл программного средства — это
- а) время продажи ПС пользователям;
 - б) период эксплуатации ПС пользователем на компьютере;
 - в) период разработки и эксплуатации ПС от замысла до полного прекращения;
 - г) общее время разработки ПС от замысла до готового продукта.
10. Какого подхода к организации процесса создания и использования ПС не существует?
- а) водопадного;
 - б) исследовательского;
 - в) сборочного;
 - г) модульного.
11. Как называется экземпляр разработанного ПС?
- а) программный продукт;
 - б) программное изделие;
 - в) готовый проект;
 - г) прикладное приложение.
12. Что не относится к критериям качества ПС?
- а) функциональность;
 - б) надежность;
 - в) стоимость;
 - г) мобильность.
13. Какой задачи для предупреждения ошибок не существует?
- а) изменение программного кода;
 - б) борьба со сложностью;
 - в) обеспечение точности перевода;
 - г) обеспечение контроля принимаемых решений.
14. Какой контроль принимаемых решений существует при разработке ПС?
- а) текущий;
 - б) внешний;
 - в) внутренний;

- г) смежный.
15. Результат процесса формулирования требований к ПС называется
- а) технологическим процессом;
 - б) внешним описанием;
 - в) файлом помощи;
 - г) внутренним описанием.
16. Какого способа разработки определения требований к ПС не существует?
- а) управляемая разработкой пользователем и разработчиком в равной степени;
 - б) управляемая пользователем разработкой;
 - в) контролируемая пользователем разработкой;
 - г) независимая от пользователя разработкой.
17. Стандартизированный набор простых свойств определения качества ПС называется:
- а) функционалами;
 - б) примитивами;
 - в) критериями;
 - г) документами.
18. Каковы подкритерии сопровождаемости?
- а) мобильность и завершенность;
 - б) точность и автономность;
 - в) изучаемость и модифицируемость;
 - г) устойчивость и эффективность.
19. Свойство, характеризующее программы ПС с точки зрения организации взаимосвязанных частей в единое целое — это
- а) модульность;
 - б) системность;
 - в) удобочитаемость;
 - г) структурированность.
20. Что не входит в функциональную спецификацию?
- а) описание внешней информационной среды;
 - б) определение внешних функций;
 - в) описание исключительных ситуаций;
 - г) описание критериев качества.
21. Какого метода контроля внешнего описания ПС не существует?
- а) статический просмотр;
 - б) смежный;
 - в) внутренний;
 - г) пользовательский.
22. Какого подхода к спецификации семантики функций не существует?
- а) табличного;
 - б) геометрического;
 - в) алгебраического;
 - г) логического.
23. Что определяет нижняя часть таблицы решений?

- а) комбинацию выполняемых действий;
 - б) комбинацию значений ситуаций;
 - в) комбинацию произвольных значений;
 - г) комбинацию состояний условий.
24. Какого вида семантики не существует?
- а) операционная;
 - б) денотационная;
 - в) формализованная;
 - г) аксиоматическая.
25. Переписывание выражения с заменой одного вхождения на другое называется
- а) композицией;
 - б) нотацией;
 - в) интерпретацией;
 - г) подстановкой.
26. Из перечисленного ODMG поддерживает наборы значений:
- а) куча
 - б) массив
 - в) список
27. Из перечисленного технология построения модели при индуктивном способе на эмпирическом этапе включает:
- а) гипотезу
 - б) интуицию
 - в) предположение
 - г) умозаключение
28. Из перечисленного, эффективность ПС рекомендуется отражать:
- а) временной эффективностью
 - б) используемостью ресурсов
29. Пункты меню для установки параметров состояния объектов могут быть, из перечисленного:
- а) взаимозависимыми
 - б) независимыми
30. В открытом состоянии окно выпадающего списка должно показывать пунктов:
- а) 3-8
 - б) 1-2
 - в) 4-10
 - г) 2-6

Вопросы к промежуточной аттестации (2, 3 семестр)

1. Сравнительный анализ и этапы развития языков программирования.
2. Методы разработки программ: процедурный, функциональный и объектно-ориентированный.
3. Понятие объекта и его свойства: инкапсуляция, наследование, полиморфизм.
4. Понятие класса. Методы классов.
5. Классификация методов.
6. Уровни инкапсуляции: private, protected, public, published.
7. Событийная модель приложений Windows.

8. Простые и сложные программы
9. Основные этапы развития ТП
10. Проблемы разработки сложных программных систем
11. Блочный-иерархический подход к созданию сложных систем
12. Жизненный цикл (ЖЦ) и этапы разработки программного обеспечения
13. Модели ЖЦ, их особенности, плюсы и минусы
14. Ускорение разработки программного обеспечения.
15. Технология RAD
16. Оценка качества процессов создания программного обеспечения
17. Понятие технологичности программного обеспечения
18. Модули и их свойства
19. Нисходящая и восходящая разработка программного обеспечения
20. Структурное и «неструктурное» программирование.
21. Средства описания структурных алгоритмов
22. Стиль оформления программы
23. Эффективность и технологичность Программирование «с защитой от ошибок»
Классификация программных продуктов по функциональному признаку
24. Основные эксплуатационные требования к программным продуктам
25. Предпроектные исследования предметной области
26. Разработка технического задания
27. Принципиальные решения начальных этапов проектирования

Тесты к экзамену (4 семестр)

1. Логическое выражение

$(N \bmod 10 \bmod 2) \text{ Or } (N \text{ Div } 10 \bmod 10 \bmod 2 = 0) \text{ Or } (N \text{ Div } 100 \bmod 2 = 0)$

должно принимать значение TRUE тогда и только тогда, когда истинно высказывание

- 1) в трёхзначном натуральном числе все цифры чётные;
- 2) в трёхзначном натуральном числе одна чётная цифра;
- 3) в трёхзначном натуральном числе две чётных цифры;
- 4) в трёхзначном натуральном числе хотя бы одна чётная цифра;
- 5) в трёхзначном натуральном числе нет чётных цифр.

2. Ошибку "Structure too large" (структура превышает максимально допустимый объём статической памяти) вызовет описание

- 1) Type Vector = Array[Byte] Of Integer; Var C : Array[1..10] Of Vector;
- 2) Var T : File Of String;
- 3) Type A = Record S : String; A, B, C : Array[10..20] Of Real End;
Var M : Array[1..5, 1..8] Of A;
- 4) Var K : Array [Byte, Byte] Of String[6];
- 5) Var S : Array[-10000..10000] Of String[2].

3. К процедурам для работы с динамическими переменными не относится

- 1) Mark;
- 2) New;
- 3) Release;
- 4) Seek;
- 5) Dispose.

4. Имеется описание

Type A = Array[0..100] Of Real; B = ^A; Var M : Array[1..5] Of B;
Для хранения массива M необходим объём памяти (байт)

- 1) 606;
- 2) 4;
- 3) 20;
- 4) 12120;
- 5) 6.

5. Фрагмент программы

```
K := 0;
While Not Eof(F) Do
Begin ReadLn(F, S); I := 1;
  While I <= Length(S) Do
  Begin If S[I] In ['А'..'Я', 'а'..'я', 'р'..'я']
  Then Begin K := K + 1;
        Delete(S, I, 1); I := I - 1
      End;
        I := I + 1
    End
  End;
End;
```

выполняет следующее действие:

- 1) удаляет из текстового файла F все русские буквы;
- 2) определяет в текстовом файле количество символов, являющихся русскими буквами;
- 3) определяет в текстовом файле количество символов, не являющихся русскими буквами;
- 4) определяет в текстовом файле количество символов;
- 5) удаляет из текстового файла F все символы, не являющиеся русскими буквами.

6. В фрагменте программы (здесь Var F : File Of Integer; I, K, Vsp : Integer;)

```
Reset(F); K := FileSize(F) - 1;
For I := 0 To K Do
Begin Seek(F, I); Read(F, Vsp); Seek(F, FileSize(F)); Write(F, Vsp) End;
```

выполняется

- 1) сортировка файла;
- 2) изменение порядка следования элементов на обратный;
- 3) дописывание в конец исходного файла полную его копию с сохранением порядка следования элементов;
- 4) дописывание в конец исходного файла полную его копию с изменением порядка следования элементов на противоположный;
- 5) не выполняется никаких действий по изменению файла.

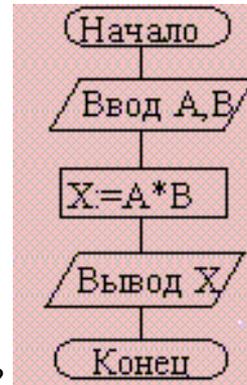
7. Имеется описание

Type Dn = (pn, vt, sr, cht, ptn, sb, vs); Mn = Set Of Dn; Var V : Mn;
и фрагмент программы

```
V := [pn..ptn] * [sr, ptn..vs] - [sb];
```

После исполнения этого фрагмента переменная V имеет значение

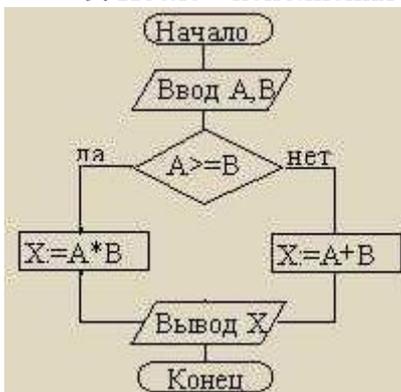
- 1) [pn..vs];
- 2) [sr, ptn];
- 3) [sb];
- 4) [];
- 5) [pn..ptn].



8. Алгоритм какого типа изображен на блок-схеме?

- 1) циклический;
- 2) разветвляющийся;
- 3) вспомогательный;
- 4) линейный;
- 5) комбинация развилки и цикла.

9. После исполнения фрагмента программы, изображенного на блок-схеме,



при $A = 5$, $B = 4$ значение X будет равно

- 1) 20;
- 2) 9;
- 3) 5;
- 4) 4;
- 5) 1.

10. В приведенном фрагменте программы (N типа LongInt, $N > 0$)

```
P := 1;
While P <= N Do
Begin
    Left := N Div (P * 10) * (P * 10);
    Right := N Mod P;
    K := ((N Mod (P * 10) Div P + 1) Mod 10) * P;
    N := Left + K + Right; P := P * 10
```

End;

натуральное число N изменяется по следующему правилу:

- 1) не изменяется;
 - 2) в каждый разряд прибавляется 1;
 - 3) из каждого разряда вычитается 1;
 - 4) в каждый разряд прибавляется 1, если значение в разряде — не девять, иначе заменяется на нуль;
 - 5) каждая девятка в десятичной записи числа заменяется на нуль.
11. Какая диаграмма служит основой для достижения взаимопонимания между программистами- профессионалами, которые разрабатывают проект, и заказчиками проекта?
- 1) диаграмма вариантов использования
 - 2) диаграмма последовательности
 - 3) Кооперативная диаграмма
 - 4) диаграмма состояний
 - 5) диаграмма действий
12. Какие диаграммы служат основой для генерации кода на целевом языке программирования?
- 1) Диаграммы вариантов использования
 - 2) Диаграммы последовательности
 - 3) Диаграммы классов
 - 4) Диаграммы состояний
13. Динамика конкретного класса может быть выражена с помощью ...
- 1) Диаграммы вариантов использования
 - 2) Диаграммы последовательности
 - 3) Диаграммы классов
 - 4) Диаграммы состояний
14. Какие программы нельзя отнести к прикладному ПО:
- 1) компиляторы и (или) интерпретаторы;
 - 2) текстовые и (или) графические редакторы;
 - 3) электронные таблицы.
15. Специфические особенности ПО как продукта:
- 1) низкие затраты при дублировании;
 - 2) универсальность;
 - 3) простота эксплуатации;
 - 4) наличие поддержки (сопровождения) со стороны разработчика.
16. Какие программы можно отнести к системному ПО:
- 1) утилиты;
 - 2) экономические программы;
 - 3) статистические программы;
 - 4) мультимедийные программы.
17. Этап, занимающий наибольшее время, в жизненном цикле программы:
- 1) сопровождение;
 - 2) проектирование;

- 3) тестирование;
- 4) программирование;
- 5) формулировка требований.

18. Этап, занимающий наибольшее время, при разработке программы:

- 1) тестирование;
- 2) сопровождение;
- 3) проектирование;
- 4) программирование;
- 5) формулировка требований.

19. Первый этап в жизненном цикле программы:

- 1) формулирование требований;
- 2) анализ требований;
- 3) проектирование;
- 4) автономное тестирование;
- 5) комплексное тестирование.

20. Один из необязательных этапов жизненного цикла программы:

- 1) оптимизация;
- 2) проектирование;
- 3) тестирование;
- 4) программирование;
- 5) анализ требований.

21. Самый большой этап в жизненном цикле программы:

- 1) эксплуатация;
- 2) изучение предметной области;
- 3) программирование;
- 4) тестирование;
- 5) корректировка ошибок.

22. Какой этап выполняется раньше:

- 1) отладка;
- 2) тестирование.

23. Какой этап выполняется раньше:

- 1) отладка;
- 2) оптимизация;
- 3) программирование;
- 4) тестирование.

24. Что выполняется раньше:

- 1) компиляция;
- 2) отладка;
- 3) компоновка;
- 4) тестирование.

25. Что выполняется раньше:

- 1) проектирование;
- 2) программирование;
- 3) отладка;

- 4) тестирование.
26. В стадии разработки программы не входит:
- 1) автоматизация программирования;
 - 2) постановка задачи;
 - 3) составление спецификаций;
 - 4) эскизный проект;
 - 5) тестирование.
27. Самый важный критерий качества программы:
- 1) работоспособность;
 - 2) надежность;
 - 3) эффективность;
 - 4) быстрое действие;
 - 5) простота эксплуатации.
28. Способы оценки качества:
- 1) сравнение с аналогами;
 - 2) наличие документации;
 - 3) оптимизация программы;
 - 4) структурирование алгоритма.
29. Существует ли связь между эффективностью и оптимизацией программы:
- 1) да;
 - 2) нет.
30. Наиболее важный критерий качества:
- 1) надежность;
 - 2) быстрое действие;
 - 3) удобство в эксплуатации;
 - 4) удобный интерфейс;
 - 5) эффективность.

Вопросы к промежуточной аттестации (4 семестр)

1. Функциональные диаграммы
2. Диаграммы потоков данных
3. Структуры данных и диаграммы отношений компонентов данных
4. UML стандартный язык описания разработки программных продуктов с использованием объектного подхода
5. Построение концептуальной модели предметной области
6. Описание поведения.
7. Системные события и операции
8. Определение отношений между объектами
9. Уточнение отношений классов Проектирование классов
10. Компоновка программных компонентов
11. Проектирование размещения программных компонентов для распределенных программных систем
12. Особенность спиральной модели разработки.
13. Реорганизация проекта
14. Психофизические особенности человека, связанные с восприятием, запоминанием и обработкой информации

15. Пользовательская и программная модели интерфейса
16. Классификации диалогов и общие принципы их разработки
17. Основные компоненты графических пользовательских интерфейсов
18. Реализация диалогов в графическом пользовательском интерфейсе
19. Пользовательские интерфейсы прямого манипулирования и их проектирование
20. Интеллектуальные элементы пользовательских интерфейсов
21. Разработка технического задания
22. Анализ предметной области, уточнение спецификаций и разработка структурной схемы
23. Проектирование интерфейса пользователя
24. Проектирование классов приложения
25. Виды контроля качества разрабатываемого программного обеспечения
26. Ручной контроль программного обеспечения
27. Структурное тестирование
28. Функциональное тестирование Тестирование модулей и комплексное тестирование
29. Оценочное тестирование
30. Отладка программного обеспечения
31. Классификация ошибок
32. Методы отладки программного обеспечения
33. Методы и средства получения дополнительной информации
34. Общая методика отладки программного обеспечения
35. Виды программных документов
36. Пояснительная записка
37. Руководство пользователя
38. Руководство системного программиста
39. Отчет по научно-исследовательской работе
40. Основные правила оформления текстовых документов

Практические задания (примеры)

1. Дана строка S , содержащая символ «*». Получить все символы, расположенные до первой «*».
2. Дана строка S . Преобразовать строку, удалив из нее все цифры.
3. Дана строка S . Группы символов, разделенные пробелами и не содержащие пробелов внутри себя, будем называть словами. Подсчитать количество слов в данной строке.
4. Заданы радиус круга R и площадь квадрата S . Определить, поместится ли квадрат в круг.
5. Вычислить наибольший общий делитель двух целых чисел.
6. Дан массив целых чисел из 12 элементов. Найти сумму первого отрицательного и первого положительного.
7. Дан массив из целых чисел $A(5 \times 5)$. Заменить нулями все элементы на главной диагонали и выше её.
8. Дана строка, состоящая из нескольких слов, между соседними словами не менее одного пробела, за последним словом точка. Во все слова, за первым символом, вставить пробел.
9. Написать программу нахождения НОД 3-х чисел, используя НОД двух чисел.
10. Написать рекурсивную процедуру вычисления факториала целого положительного числа n .
11. Пусть дан файл целых чисел. Определите, являются ли числа в файле упорядоченными по возрастанию.

12. Пусть даны два текстовых файла. Определите, равны ли они.

Индивидуальные задания (примеры)

1. Задано натуральное число a . Является ли оно чётным?
2. Задано натуральное число a . Является ли оно кратным 9?
3. Дано трехзначное число. Кратна ли сумма его цифр семи?
4. Вводится пора года. Вывести названия месяцев для этой поры года.
5. Заданы два числа. Является ли каждое из этих чисел большим 10?
6. Дано двузначное число. Является ли сумма его цифр двузначным числом кратным трём.
7. Заданы действительные x и y . Принадлежит ли точка (x, y) ветви параболы $(y=x^2)$ лежащей во второй четверти?
8. Определить, является ли данное целое число N четным трёхзначным числом.
9. Дано натуральное число N ($N < 100$), определяющее сумму денег в рублях. Дать для этого числа наименование: "рубль", "рубля", "рублей".
10. Даны действительные положительные числа x, y, z .
 - а) Выяснить, существует ли треугольник с длинами сторон x, y, z .
 - б) Если треугольник существует, то ответить – является ли он остроугольным.
11. Даны три числа. Возвести в квадрат те из них, которые положительны. Остальные оставить без изменения.
12. Дан одномерный массив чисел a_1, \dots, a_n . Найти значение максимального элемента. Если таких элементов несколько, то определить сколько их.
13. Дан одномерный массив чисел a_1, \dots, a_n . Найти их среднее арифметическое.
14. Дан одномерный массив чисел a_1, \dots, a_n . Вывести на печать элементы массива в обратном порядке строкой и столбцом.
15. Дан одномерный массив чисел a_1, \dots, a_n . Найти наибольший элемент, поставить его первым.
16. Дан одномерный массив чисел a_1, \dots, a_n . Найти сумму всех положительных и сумму всех отрицательных чисел.
17. Дан одномерный массив чисел a_1, \dots, a_n . Определить суммы всех нечетных и всех четных чисел массива.
18. Известны данные о среднемесячной температуре за год. Определить, какая была самая высокая температура зимой, весной, летом и осенью.
19. Известны данные о среднемесячной температуре за год. Определить разность между максимальной и минимальной температурами зимой и летом.
20. Дан одномерный массив чисел a_1, \dots, a_{24} , представляющий собой данные изменения температуры воздуха в течение суток. Найти среднюю температуру и отклонение от нее в 6 и 18 часов.
21. Дан одномерный массив чисел a_1, \dots, a_n . Ввести число N . Проверить, есть ли в массиве элементы, равные этому числу. Если их несколько, то определить их количество.
22. Дан одномерный массив чисел a_1, \dots, a_n . Определить количества элементов, делящихся только на 3, только на 5, одновременно на 3 и 5.

23. Дан одномерный массив чисел a_1, \dots, a_n . Найти максимальное значение элемента и его порядковый номер.
24. В одномерном массиве найти третий нечетный элемент.
25. В одномерном массиве найти сумму максимального и минимального элементов.
26. Дан одномерный массив чисел a_1, \dots, a_n . Найти сумму элементов массива, кратных трем.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Специфика формирования компетенций и их измерение определяется структурированием информации о состоянии уровня подготовки обучающихся.

Алгоритмы отбора и конструирования заданий для оценки достижений в предметной области, техника конструирования заданий, способы организации и проведения стандартизированных оценочных процедур, методика шкалирования и методы обработки и интерпретации результатов оценивания позволяют обучающимся освоить компетентностно-ориентированные программы дисциплин.

Формирование компетенций осуществляется в ходе всех видов занятий, практики, а контроль их сформированности на этапе текущей, промежуточной и итоговой аттестации.

Оценивание знаний, умений и навыков по учебной дисциплине осуществляется посредством использования следующих видов оценочных средств:

- опросы: устный, письменный;
- задания для практических занятий;
- ситуационные задания;
- контрольные работы;
- коллоквиумы;
- написание реферата;
- написание эссе;
- решение тестовых заданий;
- экзамен.

Опросы по вынесенным на обсуждение темам

Устные опросы проводятся во время практических занятий и возможны при проведении аттестации в качестве дополнительного испытания при недостаточности результатов тестирования и решения заданий. Вопросы опроса не должны выходить за рамки объявленной для данного занятия темы. Устные опросы необходимо строить так, чтобы вовлечь в тему обсуждения максимальное количество обучающихся в группе, проводить параллели с уже пройденным учебным материалом данной дисциплины и смежными курсами, находить удачные примеры из современной действительности, что увеличивает эффективность усвоения материала на ассоциациях.

Основные вопросы для устного опроса доводятся до сведения студентов на предыдущем практическом занятии.

Письменные опросы позволяют проверить уровень подготовки к практическому занятию всех обучающихся в группе, при этом оставляя достаточно учебного времени для иных форм педагогической деятельности в рамках данного занятия. Письменный опрос проводится без предупреждения, что стимулирует обучающихся к систематической подготовке к занятиям. Вопросы для опроса готовятся заранее, формулируются узко, дабы обучающийся имел объективную возможность полноценно его осветить за отведенное время.

Письменные опросы целесообразно применять в целях проверки усвояемости значительного объема учебного материала, например, во время проведения аттестации, когда необходимо проверить знания, обучающихся по всему курсу.

При оценке опросов анализу подлежит точность формулировок, связность изложения материала, обоснованность суждений.

Решение заданий (кейс-методы)

Решение кейс-методов осуществляется с целью проверки уровня навыков (владений) обучающегося по применению содержания основных понятий и терминов дисциплины вообще и каждой её темы в частности.

Обучающемуся объявляется условие задания, решение которого он излагает либо устно, либо письменно.

Эффективным интерактивным способом решения задания является сопоставления результатов разрешения одного задания двумя и более малыми группами обучающихся.

Задачи, требующие изучения значительного объема, необходимо относить на самостоятельную работу студентов, с непременным разбором результатов во время практических занятий. В данном случае решение ситуационных задач с глубоким обоснованием должно представляться на проверку в письменном виде.

При оценке решения заданий анализируется понимание обучающимся конкретной ситуации, правильность её понимания в соответствии с изучаемым материалом, способность обоснования выбранной точки зрения, глубина проработки рассматриваемого вопроса, умением выявить основные положения затронутого вопроса.

Решение заданий в тестовой форме

Проводится тестирование в течение изучения дисциплины

Не менее чем за 1 неделю до тестирования, преподаватель должен определить обучающимся исходные данные для подготовки к тестированию: назвать разделы (темы, вопросы), по которым будут задания в тестовой форме, теоретические источники (с точным указанием разделов, тем, статей) для подготовки.

При прохождении тестирования пользоваться конспектами лекций, учебниками, и иными материалами не разрешено.