

**Федеральное государственное бюджетное образовательное
учреждение высшего образования
«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА
И ГОСУДАРСТВЕННОЙ СЛУЖБЫ
ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**ИНСТИТУТ ОТРАСЛЕВОГО МЕНЕДЖМЕНТА
Факультет инженерного менеджмента
Кафедра теории и систем отраслевого управления**

УТВЕРЖДЕНА

кафедрой теории и систем отраслевого
управления

Протокол от «28» августа 2019 г.

№ 1

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.Б.24 Технологии и методы программирования

направление подготовки

27.03.05 – Инноватика

направленность (профиль) "Технологическое предпринимательство"

Квалификация

Бакалавр

Форма обучения

Очная

Год набора - 2020

Москва, 2019 г.

Авторы–составители:

Профессор кафедры теории и систем отраслевого управления, д.э.н., профессор Н.В. Новичков,

преподаватель кафедры теории и систем отраслевого управления Н.И. Пышков

Заведующий кафедрой теории и систем отраслевого управления, к.э.н., доцент С.С. Серебренников

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы
2. Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся и место дисциплины в структуре образовательной программы
3. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий и структура дисциплины
4. Материалы текущего контроля успеваемости обучающихся и фонд оценочных средств промежуточной аттестации по дисциплине
5. Методические указания для обучающихся по освоению дисциплины
6. Основная и дополнительная учебная литература, необходимая для освоения дисциплины, ресурсы информационно-телекоммуникационной сети "Интернет", включая перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине
7. Материально-техническая база, информационные технологии, используемые при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

1.1. Дисциплина Б1.Б.24 «Технологии и методы программирования» обеспечивает овладение следующей компетенцией с учетом этапа:

Код компетенции	Наименование компетенции	Код этапа освоения компетенции	Наименование этапа освоения компетенции
ОПК-3	Способность использовать информационно-коммуникационные технологии, управлять информацией с использованием прикладных программ деловой сферы деятельности, использовать компьютерные технологии и базы данных, пакеты прикладных программ управления проектами	ОПК-3.1	Способность использовать современные технологии для управления информацией и базами данных, а также управления проектами

1.2. В результате освоения дисциплины у студентов должны быть сформированы:

Код этапа освоения компетенции	Результаты обучения
ОПК-3.1	<p>на уровне знаний: истории, логики и тенденций развития технологий программирования; современных представлений о методах и технологиях программирования; стандартов в области разработки и реализации программного обеспечения; о жизненном цикле программы; методологических принципов разработки и реализации алгоритмов и программ; понятий и видов пользовательских интерфейсов; основных компонентов пользовательских интерфейсов</p> <p>на уровне умений: идентифицировать и классифицировать методы программирования; грамотно ориентироваться в существующих технологиях программирования; ориентироваться в современных моделях жизненного цикла ПО; применять знания в области жизненного цикла к организации и разработке ПО; выбирать наиболее подходящие типы пользовательских интерфейсов для решения практических задач</p> <p>на уровне навыков: выбирать модели разработки ПО; оценивать качество разработанных алгоритмов и программ; выполнять стандартные операции над данными различного типа</p>

2. Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся и место дисциплины в структуре образовательной программы

Объем дисциплины

Вид учебных занятий и самостоятельная работа		Объем дисциплины, час.	
		Всего	Семестр
			3
Очная форма обучения			
Контактная работа обучающихся с преподавателем, в том числе:		48	48
лекционного типа (Л)		16	16
лабораторные работы (практикумы) (ЛР)			
практического (семинарского) типа (ПЗ)		32	32
Самостоятельная работа обучающихся (СР)		24	24
Промежуточная аттестация	форма	экзамен	экзамен
	час.	36	36
Общая трудоемкость (час. / з.е.)		108/3	108/3

Место дисциплины в структуре ОП ВО

Дисциплина Б1.Б.24 «Технологии и методы программирования» изучается в 3 семестре очной формы обучения, общая трудоемкость дисциплины – 3 зачетные единицы, 108 часов.

Дисциплина реализуется после изучения дисциплины Б1.Б.04 «Информационные технологии» и является основой для изучения дисциплины Б1.В.04 «Информационные технологии в управлении проектами».

Форма промежуточной аттестации – экзамен.

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

Структура дисциплины

№ п/п	Наименование тем	Объем дисциплины, час.						Форма текущего контроля успеваемости*, промежуточной аттестации**
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л	ЛР	ПЗ	КСР		
Очная форма обучения								
Тема 1	Эволюция технологии программирования	12	2		6		4	О, Т
Тема 2	Основные этапы технологии программирования. Пользовательский интерфейс	12	2		6		4	О, Т
Тема 3	Базовые элементы языка C++. Функции и операторы C++	12	2		6		4	О, ЗП, Т, Д

Тема 4	Массивы и строки. Сортировки	14	4		6		4	О, ЗП, Т
Тема 5	Классы и объекты. Наследование	10	2		4		4	О, ЗП, Т, Д
Тема 6	Объектно-ориентированная реализация списков. Реализация списков с помощью библиотеки стандартных шаблонов	12	4		4		4	Т, ЗП
Промежуточная аттестация		36						Экз
Всего:		108	16		32		24	36

Примечание:

* – формы текущего контроля успеваемости: опрос (О), тестирование (Т), задания по программированию (ЗП), доклад (Д);

** - форма промежуточной аттестации: экзамен (Экз).

Содержание дисциплины

Тема 1. Эволюция технологии программирования

Неструктурированное программирование. Процедурное и модульное программирование. Объектно-ориентированное программирование. Декларативное программирование. Компонентные технологии. Перспективы развития технологий программирования.

Тема 2. Основные этапы технологии программирования. Пользовательский интерфейс

Алгоритмы и программы. Жизненный цикл программы. Постановка задачи и спецификация программы. Проектирование и реализация программы. Документирование программ.

Типы пользовательских интерфейсов. Классификация диалогов и их реализация. Основные компоненты интерфейсов.

Тема 3. Базовые элементы языка C++. Функции и операторы C++

Состав языка. Структура программы. Стандартные типы данных C++. Константы. Переменные. Организация консольного ввода/вывода данных. Операции. Выражения и преобразование типов. Примеры простейших программ.

Основные понятия. Локальные и глобальные переменные. Параметры функции. Классы памяти. Модели памяти. Примеры использования функций при решении задач.

Операторы следования. Операторы ветвления. Примеры использования операторов ветвления при решении задач. Операторы цикла. Примеры использования операторов цикла при решении задач. Операторы безусловного перехода.

Тема 4. Массивы и строки. Сортировки

Указатели. Ссылки. Одномерные массивы. Примеры использования одномерных массивов. Двумерные массивы. Примеры использования двумерных массивов. Вставка и удаление элементов в массивах.

Работа со строками в виде массивов символов. Класс string. Взаимное преобразование объектов типа string и строк в стиле C. Работа с отдельными символами. Смешанный строко-числовой ввод данных. Примеры работы со строками.

Метод «пузырька». Сортировка вставками. Сортировка посредством выбора. Алгоритм сортировки Шелла. Решение практических задач с использованием сортировок

Тема 5. Классы и объекты. Наследование

Классы и объекты: основные понятия. Конструкторы. Деструкторы. Статические члены класса. Перегрузка операций. Пример простого класса. Упражнения.

Наследование: основные понятия. Наследование конструкторов. Виртуальные функции. Абстрактные классы и чисто виртуальные функции.

Тема 6. Объектно-ориентированная реализация списков. Реализация списков с помощью библиотеки стандартных шаблонов

Основные понятия. Стек. Решение практических задач с использованием стеков. Применение исключений и шаблонов. Очередь. Решение практических задач с использованием очереди. Однонаправленный список общего вида. Решение практических задач с использованием однонаправленных списков. Двухнаправленный список. Решение практических задач с использованием двухнаправленных списков.

Класс-контейнер stack. Класс-контейнер queue. Класс-контейнер list. Решение практических задач с использованием библиотеки STL.

4. Материалы текущего контроля успеваемости обучающихся и фонд оценочных средств промежуточной аттестации по дисциплине

4.1. Формы и методы текущего контроля успеваемости обучающихся и промежуточной аттестации.

4.1.1. В ходе реализации дисциплины Б1.Б.24 «Технологии и методы программирования» используются следующие методы текущего контроля успеваемости обучающихся:

- при проведении занятий лекционного типа:
опрос, тестирование;
- при проведении занятий семинарского типа:
задания по программированию;
- при контроле результатов самостоятельной работы студентов:
тестирование, доклад, задания по программированию.

4.1.2. Экзамен проводится в письменной форме выполнением трех заданий по программированию.

4.2. Материалы текущего контроля успеваемости обучающихся

Типовые оценочные материалы по теме 1

Вопросы опроса

1. Какие этапы эволюции прошли технологии программирования?
2. Какие языки и методы программирования вы знаете?
3. Какие языки программирования называются языками высокого уровня?
4. Какая модель построения программ лежит в основе технологии процедурного программирования?
5. Каковы основные методы процедурного программирования?
6. На чем основывается концепция объектно-ориентированного программирования?
7. Каковы основные принципы объектно-ориентированного программирования?
8. Что такое компонентные технологии и CASE-технологии?
9. В чем преимущества и недостатки языков сценария?
10. Какова область применения языков параллельных вычислений?"

Вопросы тестирования

1. Что позволило программистам вместо кодов использовать мнемонические обозначения кодов операций и символические имена данных?

- A. возникновение библиотеки подпрограмм
 - B. создание совместной разработки ПО
 - C. появление ассемблеров
 - D. разработка аппаратных средств
2. Как называется совокупность методов и средств, используемых в процессе разработки программного обеспечения?
- A. технология программирования
 - B. условия программирования
 - C. операции программирования
 - D. структура программирования
3. В чем заключается метод структурного программирования процедур?
- A. сохранение текста процедуры
 - B. объединение процедур из вложенных блоков
 - C. разделение процедур на вложенные блоки
 - D. форматирование текста процедуры
 - E. использование операторов ветвления и циклов
 - F. исключение операторов ветвления и циклов
4. Какой язык программирования был создан Н. Виртом в 1971 г. для изучения процедурного программирования?
- A. BCPL
 - B. FORTRAN
 - C. C
 - D. UNIX
 - E. Pascal
5. Как называется совокупность объектов, имеющих одинаковые свойства и методы?
- A. тип
 - B. вид
 - C. атрибут
 - D. класс
6. Что означает инкапсуляция?
- A. каждый субъект объемно описывается совокупностью своих свойств и методов
 - B. каждый объект полностью описывается совокупностью своих свойств и методов
 - C. каждый предмет частично описывается совокупностью своих свойств и методов
 - D. каждый субъект выборочно описывается совокупностью своих свойств и методов
 - E. каждый объект фрагментарно описывается совокупностью своих свойств и методов
7. В рамках какого стиля программирования программа представляет собой совокупность утверждений, описывающих фрагмент предметной области или сложившуюся ситуацию?
- A. полиморфный стиль программирования
 - B. императивный стиль программирования
 - C. абстракционный стиль программирования
 - D. декларативный стиль программирования
8. Какое программирование возникло как упрощение функционального программирования для математиков и лингвистов, решающих задачи символьной обработки?
- A. многослойное
 - B. ориентированное
 - C. наследственное
 - D. логическое
9. Что является основой среды JavaBeans?
- A. компонентная субъектная модель
 - B. компонентная объектная модель

- C. межкомпонентная предметная модель
 - D. объединённая программная модель
10. Какой технологии соответствует модель взаимодействия типа «клиент — сервер»?
- A. технология CASE
 - B. технология CORBA
 - C. технология COM
 - D. технология EJB

Типовые оценочные материалы по теме 2

Вопросы опроса

1. Что называется архитектурой программного обеспечения? Каковы основные типы пользовательских интерфейсов?
2. Что такое спецификации, какие сведения они содержат?
3. Что представляет собой структурный анализ? Какие правила лежат в основе структурного анализа?
4. Какая модель построения программы используется при объектно-ориентированном подходе?
5. В чем заключается этап реализации программного обеспечения? Какие методы оценки трудоемкости разработки программного обеспечения вы знаете?
6. Какие виды организации коллектива разработчиков программного обеспечения вы знаете?
7. Какие работы выполняются в процессе обеспечения качества программного продукта? Какие факторы влияют на качество программ?
8. Какие основные группы ошибок в программных продуктах вы знаете?
9. Что подразумевается под защитным программированием?
10. Каковы основные группы документации программного обеспечения?

Вопросы тестирования

1. Какая модель жизненного цикла программного обеспечения предполагает переход на следующий этап после завершения всех операций предыдущего этапа?
 - A. каскадная модель
 - B. модель с промежуточным контролем
 - C. ядерная модель
 - D. спиральная модель
2. Модель, в которой коллектив разработчиков состоит из равноправных специалистов, занимающихся приблизительно одинаковыми задачами в рамках единого проекта:
 - A. каскадная модель
 - B. модель с промежуточным контролем
 - C. ядерная модель
 - D. спиральная модель
3. Модель, предполагающая наличие одного главного разработчика, задачей которого является создание прототипа системы:
 - A. каскадная модель
 - B. модель с промежуточным контролем
 - C. ядерная модель
 - D. спиральная модель
4. Как именуется процесс формулировки назначения программного обеспечения и основных требований к нему?
 - A. проектирование
 - B. определение спецификаций
 - C. постановка задачи
 - D. реализация

5. Как именуется процесс, предполагающий обеспечение гарантий того, что программные продукт и процессы соответствуют установленным требованиям и утверждённым планам??
 - A. проектирование
 - B. обеспечение качества
 - C. постановка задачи
 - D. реализация
6. Как называется совокупность взаимосвязанных программ, совместно решающих небольшой класс задач некоторой прикладной области?
 - A. программные системы
 - B. пакеты программ
 - C. программные комплексы
 - D. архитектура программ
7. Какой интерфейс реализует единственный сценарий работы программного обеспечения?
 - A. интерфейс со свободной навигацией
 - B. интерфейс прямого манипулирования
 - C. примитивный интерфейс
 - D. интерфейс-меню
8. Какая форма диалога предполагает общение с пользователем на естественном языке?
 - A. табличная
 - B. директивная
 - C. фразовая
 - D. все варианты верны
9. Какое окно используют для просмотра и задания различных режимов работы и необходимых параметров?
 - A. окно приложения
 - B. информационное окно
 - C. окно меню
 - D. диалоговое окно
 - E. дочернее окно
10. Как называется небольшое окно с графическим изображением?
 - A. пиктограмма
 - B. анаграмма
 - C. диаграмма
 - D. картограмма

Типовые оценочные материалы по теме 3

Вопросы опроса

1. Каково минимальное количество итераций в цикле с постусловием?
2. Назовите назначение оператора break. Где его можно применять?
3. Укажите возможности оператора goto при вложениях циклов.
4. Где и когда употребляется оператор continue?
5. Какого типа может быть значение переключающего выражения в переключателе?
6. Что называют меткой переключателя?
7. Каким оператором должна завершиться ветвь переключателя?
8. Какая конструкция вводит ветвь переключателя?
9. В каких случаях выполняется ветвь переключателя, введенная меткой со служебным словом default?
10. Приведите примеры шаблона типа в метках переключателя.
11. Почему при переключениях по типу метки переключателя могут оказаться не взаимоисключающими?

Задания по программированию

1. Разработать функцию $\min(a,b)$ для нахождения минимального из двух чисел. Вычислить с ее помощью значение выражения
$$z=\min(3x,2y)+\min(x-y,x+y)$$
2. Разработать функцию $\max(a,b)$ для нахождения максимального из двух чисел. Вычислить с ее помощью значение выражения
$$z=\max(x,2y-x)+\max(5x+3y,y)$$
3. Разработать функцию $f(x)$, которая нечетное число заменяет на 0, а четное число уменьшает в два раза. Продемонстрировать работу данной функции на примере.
4. Разработать функцию $f(x)$, которая число, кратное пяти, уменьшает в пять раз, а остальные числа увеличивает на единицу. Продемонстрировать работу данной функции на примере.
5. Разработать функцию $f(x)$, которая в двузначном числе меняет цифры местами, а остальные числа оставляет без изменения. Продемонстрировать работу данной функции на примере.
6. Написать программу, которая подсчитывает периметр квадрата, площадь которого равна a .
7. Написать программу, которая подсчитывает площадь равностороннего треугольника, периметр которого равен p .
8. Написать программу, которая подсчитывает расстояние между точками с координатами a, b и c, d .
9. Написать программу, которая подсчитывает среднее арифметическое кубов двух данных чисел.
10. Написать программу, которая подсчитывает среднее геометрическое модулей двух данных чисел.
11. Написать программу, которая подсчитывает гипотенузу прямоугольного треугольника по двум данным катетам a, b .
12. Написать программу, которая подсчитывает площадь прямоугольного треугольника по двум катетам a, b .

Вопросы тестирования

1. Как называется именованная последовательность описаний и операторов, выполняющая какое-либо законченное действие?
А. операция
В. функция
С. преобразование
D. выражение
2. Какими переменными являются все величины, описанные внутри функции, а также ее параметры?
А. фиксированными
В. глобальными
С. локальными
D. общими
3. Чем обладает блок, когда объявленные внутри блока имена доступны только внутри данного блока или блоков, вложенных в него?
А. собственной областью определенности
В. собственной областью применимости
С. собственной областью значимости
D. собственной областью видимости
4. Что является частным случаем оператора выражения, для использования, когда по синтаксису оператор требуется, а по смыслу — нет?
А. оператор инкремента «++i;»

- В. вызов функции «f(a, b);»
 - С. пустой оператор «;»
 - Д. оператор сложения с присваиванием «x+= y;»
5. Какой оператор предназначен для разветвления процесса вычислений на несколько направлений?
 - А. оператор вывода cout
 - В. оператор передачи управления break
 - С. условный оператор if
 - Д. оператор выбора switch
 6. Какой оператор используется для разветвления процесса обработки данных на два направления?
 - А. оператор выбора switch
 - В. условный оператор if
 - С. оператор ввода cin
 - Д. оператор передачи управления break
 7. Что представляет собой оператор (простой или составной):
 - А. тело цикла
 - В. модификацию цикла
 - С. инициализация цикла
 - Д. итерацию цикла
 8. Какие операторы используются для организации многократно повторяющихся вычислений?
 - А. операторы цикла
 - В. операторы выбора
 - С. операторы ветвления
 - Д. операторы следования
 9. Оператор перехода к следующей итерации цикла continue:
 - А. пропускает все операторы, оставшиеся до конца тела цикла
 - В. запускает все операторы, оставшиеся до конца тела цикла
 - С. передает управление на помеченный меткой оператор
 - Д. завершает выполнение функции и передает управление в точку ее вызова
 10. Какой формат имеет оператор безусловного перехода?
 - А. cout << i << «\t»;
 - В. return 0;
 - С. goto <метка>;
 - Д. continue;

Темы докладов

1. Арифметические операции, явные и неявные преобразования типов.
2. Условные операторы и логические выражения.
3. Операторы цикла и логические выражения.
4. Вложенные условные операторы и оператор switch.
5. Условная операция, логические операции, целые величины в качестве логических.

Типовые оценочные материалы по теме 4

Вопросы опроса

1. Что такое указатель?
2. Какие операции применяются для работы с указателями?
3. Что такое массив?
4. Что означает операция индексации для массива?
5. В чем особенности хранения многомерных массивов?
6. Как производится инициализация массивов?
7. Что такое динамический массив?

8. Какие функции применяются для управления динамическим выделением памяти?
9. Каковы особенности динамического выделения памяти для многомерных массивов?
10. Каковы особенности хранения строковых данных?

Задания по программированию

1. Дана последовательность целых чисел. Заменить все положительные элементы противоположными им числами. Задачу решить двумя способами, используя одномерный массив, а затем двумерный.
2. Дана последовательность целых чисел. Заменить все элементы, меньшие заданного числа, этим числом. Задачу решить двумя способами, используя одномерный массив, а затем двумерный.
3. Дана последовательность целых чисел. Заменить все элементы, попадающие в интервал $[a, b]$, нулем. Задачу решить двумя способами, используя одномерный массив, а затем двумерный.
4. Дана последовательность целых чисел. Заменить все отрицательные элементы, не кратные трем, противоположными им числами. Задачу решить двумя способами, используя одномерный массив, а затем двумерный.
5. Дана последовательность целых чисел. Все элементы, меньшие заданного числа, увеличить в два раза. Задачу решить двумя способами, используя одномерный массив, а затем двумерный.
6. Дан массив размером $n \times n$, элементы которого – целые числа. Для каждой строки подсчитать сумму элементов, не попадающих в заданный интервал, и записать данные в новый массив.
7. Дан массив размером $n \times n$, элементы которого – целые числа. Подсчитать сумму элементов каждой строки и записать данные в новый массив. Найти максимальный элемент нового массива.
8. Дан массив размером $n \times n$, элементы которого – целые числа. Подсчитать произведение элементов каждого столбца и записать данные в новый массив. Найти минимальный элемент нового массива.
9. Дан массив размером $n \times n$, элементы которого – целые числа. Для каждой строки найти номер первой пары неравных элементов. Данные записать в новый массив.

Вопросы тестирования

1. Какой верный формат объявления строки в виде массива символов?
A. `char ([n+1] <имя строки>);`
B. `char <имя строки> [n+1];`
C. `char [<имя строки> n+1];`
D. `<имя строки> char [n+1];`
2. Как называется последовательность символов определенной длины?
A. строка
B. список
C. таблица
D. совокупность
3. С чего начинается нумерация символов в строке?
A. с нуля
B. с буквы
C. с пробела
D. с единицы
4. Как можно объявить переменную типа string?
A. `string (<имя переменной>[значение]);`
B. `string (<значение>)[имя переменной];`
C. `string <имя переменной>(<значение>);`
D. `string (<значение>)<имя переменной>;`

5. Функция `strcpy` не может принимать в качестве второго аргумента объект типа `string` и `C++`:
 - A. автоматически преобразует строки `C` в объекты `string`
 - B. не выполняет автоматического преобразования объектов `string` в строки `C`
 - C. иногда автоматически не преобразует строки `C` в объекты `string`
 - D. автоматически преобразует объекты `string` в строки `C`
6. Как правильно произвести копирование объекта `string` в строку `C`?
 - A. `strcpy(s1, s2.c_str());`
 - B. `strcpy(s1, c_str(s2));`
 - C. `strcpy(c_str().s1, s2);`
 - D. `strcpy(s1, c_str().s2);`
7. Какая функция возвращает значение `true`, если `c` — знак пунктуации?
 - A. `ispunct(c)`
 - B. `isalnum(c)`
 - C. `tolower(c)`
 - D. `isxdigit(c)`
8. Какая функция возвращает значение `true`, если `c` — буква?
 - A. `isalpha(c)`
 - B. `isupper(c)`
 - C. `islower(c)`
 - D. `isdigit(c)`
9. Какая операция применяется при смешанном вводе данных для ввода числовой информации?
 - A. операция `<<`
 - B. операция `cout`
 - C. операция `>>`
 - D. операция `\n`
10. Что применяется при смешанном вводе данных для ввода строковых данных?
 - A. функция `name`
 - B. функция `cout`
 - C. функция `cin`
 - D. функция `getline`

Типовые оценочные материалы по теме 5:

Вопросы опроса

1. Назовите модификаторы класса, применяемые при отсутствии наследования.
2. Назовите разновидности членов класса.
3. Какие элементы являются обязательными в объявлении нестатического поля?
4. Когда выполняется инициализация нестатических полей?
5. Каков статус доступа нестатического поля при отсутствии в его объявлении модификаторов доступа?
6. Можно ли объявить статическое поле с типом класса, которому оно принадлежит?
7. В каком случае в классе могут одновременно присутствовать одноименные статический и нестатический методы?
8. В каких случаях телом нестатического метода может быть пустой оператор?
9. В каком случае конструктор умолчания (конструктор без параметров) создается автоматически?
10. Назовите возможные применения ссылки `this`.

Задания по программированию

1. Создать класс `Time` с полями `hours`, `minutes`, `seconds`, представляющими собой часы, минуты и секунды. Данный класс должен позволять выводить информацию о текущем времени, вычислять время через заданное количество часов, минут и секунд, а также

- вычислять время, прошедшее между двумя заданными моментами. Детальную структуру класса продумайте самостоятельно.
2. Создать класс Data с полями year, month, day, представляющими собой год, месяц и день. Данный класс должен позволять выводить информацию о текущей дате, вычислять дату следующего дня, дату предыдущего дня, дату дня через заданное количество лет, месяцев и дней, а также вычислять временной отрезок (в годах, месяцах и днях) между двумя заданными датами. Детальную структуру класса продумайте самостоятельно.
 3. Создать абстрактный класс Figure с функциями вычисления площади и периметра, а также функцией, выводящей информацию о фигуре на экран. Создать производные классы: Rectangle (прямоугольник), Circle (круг), Triangle (треугольник). Создать массив n фигур и вывести полную информацию о фигурах на экран.
 4. Создать абстрактный класс Goods с функциями, позволяющими вывести на экран информацию о товаре, а также определить, соответствует ли она искомому типу. Создать производные классы: Toy (название, цена, производитель, материал, возраст, на который рассчитана), Book (название, автор, цена, издательство, возраст, на который рассчитана), SportsEquipment (название, цена, производитель, возраст, на который рассчитан). Создать базу (массив) из n товаров, вывести полную информацию из базы на экран, а также организовать поиск товаров определенного типа.
 5. Создать абстрактный класс TelephoneDirectory с функциями, позволяющими вывести на экран информацию о записях в телефонном справочнике, а также определить соответствие записи критерию поиска. Создать производные классы: Persona (фамилия, адрес, номер телефона), Organization (название, адрес, телефон, факс, контактное лицо), Friend (фамилия, адрес, номер телефона, дата рождения). Создать базу (массив) из n записей, вывести полную информацию из базы на экран, а также организовать поиск в базе по фамилии.

Вопросы тестирования

1. Что определяют в теле класса?
 - A. метки и функции
 - B. операторы и синтаксис
 - C. данные и операции
 - D. типы и алгоритмы
2. Для чего предназначен конструктор?
 - A. для того, чтобы присвоить каждому члену-данному конечное значение
 - B. для того, чтобы присвоить каждому члену-данному нулевое значение
 - C. для того, чтобы присвоить каждому члену-данному фиксированное значение
 - D. для того, чтобы присвоить каждому члену-данному начальное значение
3. Какая существует специальная член-функция, которая вызывается неявно каждый раз, когда создается новый объект класса?
 - A. конструктор
 - B. деструктор
 - C. оператор
 - D. итератор
4. Как называется процесс создания новых классов на основе уже существующих классов?
 - A. производство
 - B. продлевание
 - C. наследование
 - D. преследование
5. Если в базовом классе нет конструкторов, то в производном классе:
 - A. конструктор всегда необходимо описывать — он не будет создан явным образом
 - B. конструктор можно не описывать — он будет создан неявным образом

- C. конструктор необходимо описывать и он будет создан неявным образом
 - D. конструктор можно не описывать — он будет создан явным образом
6. Если в базовом классе конструктор задается явным образом, то производный класс должен иметь собственный конструктор, в котором:
 - A. явно вызывается конструктор базового класса
 - B. явно вызывается конструктор производного класса
 - C. неявно вызывается конструктор производного класса
 - D. неявно вызывается конструктор базового класса
 7. Когда в базовом классе объявляется хотя бы одна виртуальная функция, для всех полиморфных классов (базового и всех его производных):
 - A. выводится матрица виртуальных функций
 - B. генерируется список виртуальных функций
 - C. создается единая таблица виртуальных функций
 - D. удаляется устаревшая база виртуальных функций
 8. Что позволяет реализовать механизм позднего связывания, когда требуемая реализация функции выбирается на этапе выполнения программы?
 - A. использование классовых функций
 - B. использование программных функций
 - C. использование базовых функций
 - D. использование виртуальных функций
 9. Как называется виртуальная функция, не имеющая тела?
 - A. исходная виртуальная функция
 - B. подобно виртуальная функция
 - C. нулевая виртуальная функция
 - D. чисто виртуальная функция
 10. Как называется класс, в котором есть хотя бы одна чистая виртуальная функция?
 - A. неопознанный класс
 - B. наследственный класс
 - C. абстрактный класс
 - D. предварительный класс

Темы доклада

1. Использование спецификаторов доступа при наследовании.
2. Конструкторы производного класса.
3. Перегрузка функций в производном классе.
4. Иерархия классов. Абстрактный базовый класс.
5. Множественное наследование. Пример использования.

Типовые оценочные материалы по теме 6

Вопросы тестирования

1. Как называется список, если каждый его элемент содержит указатель на следующий элемент в списке и указатель на предыдущий элемент?
 - A. прямонаправленный
 - B. однонаправленный
 - C. двусвязный
 - D. односвязный
2. Что является частным случаем списка, все операции с которым выполняются с одного конца?
 - A. указатель
 - B. сортировка
 - C. порядок
 - D. стек
3. Какие поля содержит базовый элемент стека?

- A. поле *inf*
 - B. поле *head*
 - C. поле *next*
 - D. поле *NULL*
4. Извлечь элемент из стека или посмотреть значение верхнего элемента стека можно:
 - A. только тогда, когда стек не дополнен
 - B. только тогда, когда стек переполнен
 - C. только тогда, когда стек пуст
 - D. только тогда, когда стек не пуст
 5. Сколько у очереди указателей?
 - A. два указателя
 - B. нет указателей
 - C. один указатель
 - D. три указателя
 6. Что является частным случаем списка, добавление элементов в который выполняется в один конец, а выборка и просмотр осуществляются с другого конца?
 - A. контейнер
 - B. очередь
 - C. указание
 - D. стек
 7. В общем случае список можно представить в виде линейной связанной структуры:
 - A. с обязательным фиксированным числом «точек доступа»
 - B. только с двумя «точками доступа»
 - C. с произвольным количеством «точек доступа»
 - D. только с одной «точкой доступа»
 8. Вставка в пустой однонаправленный список — это вставка элемента в начало списка, у которого:
 - A. *head* = *NULL*
 - B. *prev* = *NULL*
 - C. *inf* = *NULL*
 - D. *next* = *NULL*
 9. В общем случае базовый элемент списка может содержать:
 - A. ни одного информационного поля
 - B. только два информационных поля
 - C. только одно информационное поле
 - D. несколько информационных полей
 10. Что добавляется в базовый элемент двунаправленного списка по сравнению с однонаправленным?
 - A. *tail*
 - B. *head*
 - C. *next*
 - D. *inf*
 - E. *prev*

Задания по программированию

1. Дан файл, содержащий информацию о сотрудниках фирмы: фамилия, имя, отчество, пол, возраст, размер заработной платы. За один просмотр файла напечатать элементы файла в следующем порядке: сначала все данные о мужчинах, потом все данные о женщинах, сохраняя исходный порядок в каждой группе сотрудников.
2. Дан файл, содержащий информацию о сотрудниках фирмы: фамилия, имя, отчество, пол, возраст, размер заработной платы. За один просмотр файла напечатать элементы файла в следующем порядке: сначала все данные о сотрудниках младше 30 лет, потом

- данные об остальных сотрудниках, отсортировав данные о сотрудниках в каждой группе по размеру заработной платы
3. Создать список из целых чисел. Подсчитать количество отрицательных элементов, создав из них новый список. Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов List и DoubleLinkedList можно расширить. Ввод-вывод данных файловый.
 4. Создать список из целых чисел. Подсчитать сумму положительных элементов. Создать из положительных элементов новый список. Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов List и DoubleLinkedList можно расширить. Ввод-вывод данных файловый.
 5. Создать список из целых чисел. Найти среднее арифметическое значение четных элементов списка. Создать из четных элементов новый список. Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов List и DoubleLinkedList можно расширить. Ввод-вывод данных файловый.
 6. Создать список из слов. Подсчитать, сколько слов начинается на данную букву. Создать из них новый список, удалив их из исходного списка. Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов List и DoubleLinkedList можно расширить. Ввод-вывод данных файловый.
 7. Написать программу «Моя записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по разным полям. Хранение данных организовать с применением контейнерного класса тар или multimap.
 8. Написать программу «Моя записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по разным полям. Хранение данных организовать с применением контейнерного класса list.

4.3. Оценочные средства для промежуточной аттестации

4.3.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Код компетенции	Наименование компетенции	Код этапа освоения компетенции	Наименование этапа освоения компетенции
ОПК-3	Способность использовать информационно-коммуникационные технологии, управлять информацией с использованием прикладных программ деловой сферы деятельности, использовать компьютерные технологии и базы данных, пакеты прикладных программ управления проектами	ОПК-3.1	Способность использовать современные технологии для управления информацией и базами данных, а также управления проектами

4.3.2 Показатели и критерии оценивания компетенций на различных этапах их формирования

Этап освоения компетенции	Показатель оценивания	Критерий оценивания
ОПК-3.1	осуществляет выбор набора количественных и качественных показателей анализа рынка удовлетворения собственных нужд организации и нужд среды в зависимости от видов юридических лиц	<p>на уровне знаний: истории, логики и тенденций развития технологий программирования; современных представлений о методах и технологиях программирования; стандартов в области разработки и реализации программного обеспечения; о жизненном цикле программы; методологических принципов разработки и реализации алгоритмов и программ; понятий и видов пользовательских интерфейсов; основных компонентов пользовательских интерфейсов</p> <p>на уровне умений: идентифицировать и классифицировать методы программирования; грамотно ориентироваться в существующих технологиях программирования; ориентироваться в современных моделях жизненного цикла ПО; применять знания в области жизненного цикла к организации и разработке ПО; выбирать наиболее подходящие типы пользовательских интерфейсов для решения практических задач</p> <p>на уровне навыков: выбирать модели разработки ПО; оценивать качество разработанных алгоритмов и программ; выполнять стандартные операции над данными различного типа</p>

4.3.3 Типовые контрольные задания или иные материалы (типовые оценочные материалы), необходимые для оценки знаний, умений, навыков и (или)

опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Типовые варианты билетов

Вариант 1

1. Дана последовательность целых чисел. Подсчитать среднее арифметическое элементов. Задание выполнить двумя способами, используя одномерный массив, а затем двумерный.
2. Написать программу, которая подсчитывает периметр прямоугольного треугольника по двум катетам a , b
3. Создать список из целых чисел. После каждого элемента, равного x , вставить элемент, равный y . Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов `List` и `DoubleLinkedList` можно расширить. Ввод-вывод данных файловый.

Вариант 2

1. Дана последовательность целых чисел. Подсчитать среднее арифметическое отрицательных элементов. Задание выполнить двумя способами, используя одномерный массив, а затем двумерный.
2. Написать программу, которая подсчитывает ребро куба, площадь полной поверхности которого равна s
3. Создать список из целых чисел. Исключить из списка все элементы, равные x . Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов `List` и `DoubleLinkedList` можно расширить. Ввод-вывод данных файловый.

Вариант 3

1. Дана последовательность целых чисел. Подсчитать количество нечетных элементов. Задание выполнить двумя способами, используя одномерный массив, а затем двумерный.
2. Написать программу, которая подсчитывает ребро куба, объем которого равен v
3. Создать список из целых чисел. Заменить каждую последовательность. Задачу выполнить в четырех вариантах: со стеком, очередью, списком общего вида в односвязной и двусвязной реализации. При необходимости функциональные возможности классов-шаблонов `List` и `DoubleLinkedList` можно расширить. Ввод-вывод данных файловый.

Шкала оценивания

Оценка	Требования к знаниям
«отлично»	Оценка «отлично» выставляется, если студент демонстрирует: знание: истории, логики и тенденций развития технологий программирования; современных представлений о методах и технологиях программирования; стандартов в области разработки и реализации программного обеспечения; о жизненном цикле программы;

	<p>методологических принципов разработки и реализации алгоритмов и программ; понятий и видов пользовательских интерфейсов; основных компонентов пользовательских интерфейсов</p> <p>умение: идентифицировать и классифицировать методы программирования; грамотно ориентироваться в существующих технологиях программирования; ориентироваться в современных моделях жизненного цикла ПО; применять знания в области жизненного цикла к организации и разработке ПО; выбирать наиболее подходящие типы пользовательских интерфейсов для решения практических задач</p> <p>навыки: выбирать модели разработки ПО; оценивать качество разработанных алгоритмов и программ; выполнять стандартные операции над данными различного типа</p>
«хорошо»	<p>Оценка «отлично» выставляется, если студент демонстрирует:</p> <p>знание: истории, логики и тенденций развития технологий программирования; современных представлений о методах и технологиях программирования; стандартов в области разработки и реализации программного обеспечения; о жизненном цикле программы; методологических принципов разработки и реализации алгоритмов и программ; понятий и видов пользовательских интерфейсов; основных компонентов пользовательских интерфейсов</p> <p>умение: идентифицировать и классифицировать методы программирования; грамотно ориентироваться в существующих технологиях программирования; ориентироваться в современных моделях жизненного цикла ПО; применять знания в области жизненного цикла к организации и разработке ПО; выбирать наиболее подходящие типы пользовательских интерфейсов для решения практических задач</p>
«удовлетворительно»	<p>Оценка «отлично» выставляется, если студент демонстрирует:</p> <p>знание: истории, логики и тенденций развития технологий программирования; современных представлений о методах и технологиях программирования; стандартов в области разработки и реализации программного обеспечения;</p>

	о жизненном цикле программы; методологических принципов разработки и реализации алгоритмов и программ; понятий и видов пользовательских интерфейсов; основных компонентов пользовательских интерфейсов навыки: выбирать модели разработки ПО; оценивать качество разработанных алгоритмов и программ; выполнять стандартные операции над данными различного типа
«неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

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

Студент должен выполнить все задания и мероприятия, предусмотренные программой дисциплины (по формам текущего контроля). В случае наличия учебной задолженности студент отрабатывает пропущенные занятия в соответствии с требованиями. Оценка студента носит комплексный характер и определяется:

- ответом на экзамене;
- учебными достижениями в семестровый период.

Опрос

В ходе текущей аттестации проверяется знание обучающимися основных понятий по теме, необходимых для дальнейшего освоения дисциплины. Выбранная форма контроля способствует формированию навыка краткого и систематичного устного изложения изученного материала. При оценке ответов в первую очередь учитывается уровень теоретической подготовки студента (владение категориальным аппаратом).

Шкала оценивания

Оценка	Требования к результатам
«зачтено»	Оценка «зачтено» выставляется, если студентом: <ul style="list-style-type: none"> • продемонстрировано отличное знание изученного материала и владение категориальным аппаратом; • дан правильный ответ на вопрос с использованием профессиональной лексики и терминологии.
«не зачтено»	Оценка «не зачтено» выставляется студенту, если студентом содержание темы раскрыто фрагментарно и имеются существенные пробелы в знаниях категориального аппарата

Тестирование

Каждый вариант теста состоит из 10 вопросов. Все тестовые вопросы предполагают варианта 3-8 ответа. Обучающемуся необходимо внимательно прочитать

вопрос и выбрать правильный ответ. В зависимости от уточнения в вопросе, правильных ответов может быть от 1 до 4.

Шкала оценивания для тестирования

Оценка	Требования к результатам
«зачтено»	Оценка «зачтено» выставляется, если студентом даны верные ответы на 70% и более вопросов
«не зачтено»	Оценка «не зачтено» выставляется студенту, если студентом даны верные ответы менее, чем на 70% вопросов

Доклад

Выбранная форма контроля способствует формированию навыка систематического устного изложения самостоятельно изученного и структурированного материала.

Шкала оценивания

Оценка	Требования к результатам
«зачтено»	Оценка «зачтено» выставляется, если студентом: <ul style="list-style-type: none"> • подготовлен доклад, который содержит полную, понятную информацию по заданной теме • продемонстрировано свободное владение содержанием, ясно и грамотно изложен материал • сделаны предусмотренные темой выводы и обобщения • свободно и корректно даны ответы на вопросы и замечания аудитории
«не зачтено»	Оценка «не зачтено» выставляется студенту, если студентом: <ul style="list-style-type: none"> • подготовлен доклад, который содержит неполную или неактуальную информацию по заданной теме • не продемонстрировано свободное владение содержанием, ясно и грамотно изложен материал • некорректно даны ответы на вопросы и замечания аудитории

Задания по программированию

Задания по программированию выполняются студентами в аудитории под контролем преподавателя или самостоятельно.

Шкала оценивания

Оценка	Требования к результатам
«зачтено»	Оценка «зачтено» выставляется, если студентом: <ul style="list-style-type: none"> • получен правильный ответ не менее, чем на 50% задач • используемые функции, операторы, методы соответствуют заданию
«не зачтено»	Оценка «не зачтено» выставляется студенту, если студентом: <ul style="list-style-type: none"> • даны верные ответы менее, чем на 50% задач; • используются функции, операторы, методы не соответствующие заданию

Экзамен проводится в письменной форме выполнением трех заданий по программированию.

Процедура проведения экзамена.

Обучающийся для сдачи экзамена предъявляет экзаменатору свою зачетную книжку, получает чистые маркированные листы бумаги для выполнения заданий по

программированию. Каждый билет состоит из 3 заданий.

Время подготовки ответа: 20 минут.

На выполнение заданий студенту отводится 90 минут. После проверки преподавателем ответов (решений) на каждое задание студенту могут быть заданы дополнительные уточняющие вопросы. В случае если студент при ответе допустил несущественные неточности, ему могут быть заданы дополнительные вопросы по темам курса.

Результат по сдаче экзамена объявляется студентам после проверки ответа экзаменатором, вносится в аттестационную ведомость и в зачетную книжку.

Оценка «неудовлетворительно» проставляется только в ведомости.

5. Методические указания для обучающихся по освоению дисциплины

Процесс обучения по дисциплине Б1.Б.24 «Технологии и методы программирования» включает следующие основные виды занятий:

1. лекции;
2. практические занятия;
3. самостоятельная работа.

На лекциях студенты изучают основные этапы, методы и технологии разработки программного обеспечения, синтаксис и базовые инструменты языка C++, знакомятся с наиболее известными работами ученых и существующими практическими разработками в данной области, закрепляя полученные знания на практических занятиях. С целью обеспечения успешного обучения студенту необходимо готовиться к каждой лекции, т.к. она является важнейшей формой организации учебного процесса, поскольку знакомит с новым учебным материалом, разъясняет учебные элементы, трудные для понимания, систематизирует учебный материал, ориентирует в учебном процессе.

Подготовку к лекции рекомендуется проводить по следующему плану:

1. внимательно прочитайте материал предыдущей лекции;
2. узнайте тему предстоящей лекции (по тематическому плану, по информации лектора);
3. ознакомьтесь с учебным материалом по учебнику и учебным пособиям;
4. постарайтесь уяснить место изучаемой темы в своей профессиональной подготовке;
5. запишите возможные вопросы, которые вы зададите лектору на лекции.

Подготовку к практическому занятию рекомендуется проводить по следующему плану:

1. внимательно прочитайте материал лекций, относящихся к данному семинарскому занятию, ознакомьтесь с учебным материалом по учебнику и учебным пособиям;
2. выпишите основные термины;
3. законспектируйте главы из основных источников литературы, соответствующие изучаемой теме;
4. уясните, какие учебные элементы остались для вас неясными и постарайтесь получить на них ответ заранее (до семинарского занятия) во время текущих консультаций преподавателя;
5. готовиться можно индивидуально, парами или в составе малой группы, последние являются эффективными формами работы.

Получение углубленных знаний по изучаемой дисциплине достигается за счет дополнения часов аудиторной нагрузки самостоятельной работой студентов, которая выражается в анализе дополнительной литературы по учебной дисциплине по отдельным темам учебной программы.

Вопросы для самостоятельной подготовки к занятиям лекционного, практического (семинарского) типов

При изучении дисциплины предусматривается обеспечение гармоничной взаимосвязи между аудиторной и самостоятельной работой студентов, для чего в рамках курса предлагается набор активных и интерактивных методов занятий в развитие сюжетов, рассмотренных в рамках лекций и практических занятий.

При подготовке к занятиям студенту рекомендуется обратиться к конспекту лекции по соответствующей теме, а также изучить следующие вопросы:

Наименование	Наименование тем и/или разделов	Вопросы, выносимые на самостоятельное изучение
Тема 1	Эволюция технологии программирования	<ol style="list-style-type: none"> 1. Декларативное программирование. 2. Компонентные технологии. 3. Перспективы развития технологий программирования. <p>Технологии и методы программирования : учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2020. — 235 с.</p>
Тема 2	Основные этапы технологии программирования. Пользовательский интерфейс	<ol style="list-style-type: none"> 1. Типы пользовательских интерфейсов. 2. Классификация диалогов и их реализации 3. Основные компоненты интерфейсов <p>Технологии и методы программирования : учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2020. — 235 с.</p>
Тема 3	Базовые элементы языка C++. Функции и операторы C++	<ol style="list-style-type: none"> 1. Локальные и глобальные переменные. 2. Параметры функции. 3. Классы памяти. 4. Модели памяти. <p>Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2020.</p>
Тема 4	Массивы и строки. Сортировки	<ol style="list-style-type: none"> 1. Метод «пузырька». 2. Сортировка вставками. 3. Сортировка посредством выбора. 4. Алгоритм сортировки Шелла. <p>Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2020.</p>
Тема 5	Классы и объекты. Наследование	<ol style="list-style-type: none"> 1. Наследование: основные понятия. 2. Наследование конструкторов. 3. Виртуальные функции. 4. Абстрактные классы и чисто виртуальные функции. <p>Огнева, М. В. Программирование на языке C++:</p>

		практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2020.
Тема 6	Объектно-ориентированная реализация списков. Реализация списков с помощью библиотеки стандартных шаблонов	1. Класс-контейнер stack. 2. Класс-контейнер queue. 3. Класс-контейнер list. Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2020.

Подготовка к промежуточной аттестации

На первом занятии преподаватель информирует обучающихся о применяемой системе текущего контроля успеваемости и форме промежуточной аттестации.

Во время последующих аудиторных занятий – доводит до студентов информацию о результатах текущего контроля успеваемости.

К промежуточной аттестации необходимо готовится целенаправленно, регулярно, систематически и с первых дней обучения по данной дисциплине. Попытки освоить дисциплину в период зачётно-экзаменационной сессии, как правило, показывают не удовлетворительные результаты. В самом начале изучения учебной дисциплины познакомьтесь со следующей учебно-методической документацией:

1. программой дисциплины;
2. перечнем знаний и умений, которыми студент должен владеть;
3. тематическими планами лекций, семинарских занятий;
4. контрольными мероприятиями;
5. учебником, учебными пособиями по дисциплине, а также электронными ресурсами;
6. типовым вариантом задания к промежуточной аттестации.

После этого у студента должно сформироваться четкое представление об объеме и характере получаемых знаний и умений по дисциплине. Систематическое выполнение учебной работы на лекциях и практических занятиях позволит успешно освоить дисциплину и создать хорошую базу для прохождения промежуточной аттестации.

6. Основная и дополнительная учебная литература, необходимая для освоения дисциплины, ресурсы информационно-телекоммуникационной сети "Интернет", включая перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1 Основная литература

1. Гниденко, И. Г. Технологии и методы программирования: учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва: Издательство Юрайт, 2020. — 235 с. — (Высшее образование). — ISBN 978-5-534-02816-4. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450999>
2. Огнева, М. В. Программирование на языке C++: практический курс: учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва: Издательство Юрайт, 2020. — 335 с. — (Высшее образование). — ISBN 978-5-534-05123-0. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454165>

6.2. Дополнительная литература

1. Зыков, С. В. Программирование: учебник и практикум для вузов / С. В. Зыков. — Москва: Издательство Юрайт, 2020. — 320 с. — (Высшее образование). —

ISBN 978-5-534-02444-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450832>

2. Федоров, Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для вузов / Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2020. — 161 с. — (Высшее образование). — ISBN 978-5-534-10971-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454100>

6.3. Учебно-методическое обеспечение самостоятельной работы

1. Гниденко, И. Г. Технологии и методы программирования: учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва: Издательство Юрайт, 2020. — 235 с. — (Высшее образование). — ISBN 978-5-534-02816-4. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450999>

2. Огнева, М. В. Программирование на языке C++: практический курс: учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва: Издательство Юрайт, 2020. — 335 с. — (Высшее образование). — ISBN 978-5-534-05123-0. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454165>

6.4. Нормативные правовые документы

Не используются.

6.5. Интернет-ресурсы

1. <http://nlr.ru/> - Российская национальная библиотека
2. <https://integrum.ru/> - Национальная электронная библиотека
3. <https://www.rsl.ru/> - Российская государственная библиотека
4. <https://www.rbc.ru/> - Информационное агентство «РосБизнесКонсалтинг»
5. <http://www.consultant.ru/> - Консультант плюс
6. <http://www.garant.ru/> - Гарант
7. <https://www.economist.com/> - журнал The Economist
8. <https://www.ft.com/> - газета The Financial Times
9. <https://www.forbes.com/manufacturing/> - Новости бизнеса (производство)
10. <https://www.forbes.com/innovation/> - Новости бизнеса (инновации)
11. <https://asi.ru/> - Агентство стратегических инициатив
12. <https://www.rvc.ru/> - государственный фонд фондов Институт развития Российской Федерации
13. <https://www.kommersant.ru/> - газета Коммерсантъ
14. <https://www.vedomosti.ru/> - газета Ведомости
15. <https://secretmag.ru/> - журнал Секрет фирмы
16. www.google.com / - Поисковая система
17. www.rambler.ru / - Поисковая система
18. www.yandex.ru / - Поисковая система

6.6. Иные источники

Не используются.

7. Материально-техническая база, информационные технологии, используемые при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Помещения представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и

промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, обеспечивающие тематические иллюстрации, соответствующие примерным программам дисциплин, рабочим учебным программам дисциплин.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечены доступом в электронную информационно-образовательную среду Академии.

Академия обеспечена необходимым комплектом лицензионного и свободно распространяемого программного обеспечения: MS Windows, MS Office.

Обучающимся обеспечен доступ (удаленный доступ), в том числе в случае применения электронного обучения, дистанционных образовательных технологий, к современным профессиональным базам данных и информационным справочным системам.