

Частное образовательное учреждение высшего образования  
«САНКТ-ПЕТЕРБУРГСКИЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ  
УПРАВЛЕНИЯ И ЭКОНОМИКИ»

РАССМОТРЕНО И ОДОБРЕНО

УТВЕРЖДАЮ

На заседании кафедры  
информационных технологий и  
математики  
Протокол № 9 от 25.05.2023 г.

Первый проректор  
С.В. Авдашкевич  
28.06.2023

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дисциплина:	Б1.В.05 Объектно-ориентированное программирование
Направление подготовки:	09.03.03 Прикладная информатика
Направленность (профиль):	Прикладная информатика в экономике
Уровень высшего образования:	Бакалавриат
Форма обучения:	очная, заочная
Разработчики:	Кандидат экономических наук, доцент Щипанов Е. Ф. Кандидат технических наук, доцент Дагаев А. В.

Санкт-Петербург  
2023

### 1. Цели и задачи дисциплины:

#### Цель освоения дисциплины:

овладение базовыми навыками объектно-ориентированного программирования, необходимыми для работы над крупными программными проектами.

#### Задачи дисциплины:

- изучить теоретические основы и современные информационные технологии анализа, проектирования и разработки программного обеспечения;
- научиться проектировать и разрабатывать различные виды программного обеспечения на основе объектно-ориентированного подхода;
- получить представление о библиотеках классов и инструментальных средствах, применяемых при разработке программного обеспечения

### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы высшего образования

Планируемые результаты освоения ОП ВО (код и содержание компетенций)	Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Примечание
ПК-2 Способен разрабатывать и адаптировать прикладное программное обеспечение	ПК-2.1 Знать инструменты и методы модульного тестирования; инструменты и методы прототипирования пользовательского интерфейса; инструменты и методы тестирования нефункциональных и функциональных характеристик ИС; методы оценки качества программных систем; основы программирования; основы современных операционных систем; основы современных систем управления базами данных; основы управления взаимоотношениями с клиентами и заказчиками (CRM); программные средства и платформы инфраструктуры информационных технологий организаций; современные стандарты информационного взаимодействия систем; современные структурные языки программирования; теория тестирования; языки программирования и работы с базами данных; языки современных бизнес-приложений.	06.015 Профессиональный стандарт «Специалист по информационным системам» 06.022 Профессиональный стандарт «Системный аналитик»
	ПК-2.2 Уметь алгоритмизировать деятельность; кодировать на языках программирования; тестировать результаты прототипирования.	
	ПК-2.3 Владеть навыками проведения анализа результатов тестов; координирования и проведения оценки готовых систем; обучения участников рабочей группы методике оценки готовых систем; оформления отчета о степени соответствия готовых систем требованиям; подготовки методике оценки готовых систем на соответствие требованиям; принятия решения о пригодности архитектуры; разработки прототипа ИС в соответствии с требованиями; осуществления сбора, обработки и анализа результатов оценки готовых систем на соответствие требованиям; согласования пользовательского интерфейса с заказчиком; тестирования прототипа ИС на проверку корректности архитектурных решений.	

Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Планируемые результаты обучения по дисциплине
ПК-2.1. Знать инструменты и методы модульного тестирования; инструменты и методы прототипирования пользовательского ин-терфейса; инструменты и методы тестирования нефункциональных и функциональных характеристик ИС; методы оценки качества программных систем; основы программирования; ос-новы современных операционных систем; основы современных систем управления базами данных; основы управления взаимоотношениями с клиентами и заказчиками (CRM); программ-ные средства и платформы инфраструктуры информационных технологий организаций; современные стандарты информаци-онного взаимодействия систем; современные структурные язы-ки программирования; теория тестирования; языки программи-рования и работы с базами данных; языки современных бизнес-приложений.	Знать основы объектно-ориентированного программирования.
ПК-2.2. Уметь алгоритмизировать деятельность; кодировать на языках программирования; тестировать результаты прототипирова-ния.	Уметь алгоритмизировать деятельность; кодировать на языках объектно-ориентированного программирования; тестировать результаты прототипирования.
ПК-2.3. Владеть навыками проведения анализа результатов тестов; ко-ординирования и проведения оценки готовых систем; обучения участников рабочей группы методике оценки готовых систем; оформления отчета о степени соответствия готовых систем требо-ваниям; подготовки методики оценки готовых систем на со-ответствие требованиям; принятия решения о пригодности ар-хитектуры; разработки прототипа ИС в соответствии с требо-ваниями; осуществления сбора, обработки и анализа результа-тов оценки готовых систем на соответствие требованиям; согла-сования пользовательского интерфейса с заказчиком; тестиро-вания прототипа ИС на проверку корректности архитектурных решений.	Владеть навыками проведения анализа результатов тестов; координирования и проведения оценки готовых систем; обработки и анализа результатов оценки готовых систем на соответствие требованиям.

### 3. Содержание, объем дисциплины и формы проведения занятий

№ п/п	Наименование темы дисциплины	Компетенции	Оценочные средства текущего контроля		
			ЗНАТЬ	УМЕТЬ	ВЛАДЕТЬ
			ПК-2.1	ПК-2.2	ПК-2.3
1	Объектно-ориентированный подход в программировании.	ПК-2	Доклад, сообщение/ Реферат №1 (10)	Собеседование, опрос/ Контрольная работа №1 (10)	Деловая и (или) ролевая игра/Кейс-задача №1 (20)
2	Классы и объекты.	ПК-2	Доклад, сообщение/ Реферат №2 (10)	Собеседование, опрос/ Контрольная работа №2 (10)	Деловая и (или) ролевая игра/Кейс-задача №1 (20)
3	Наследование, базовый и производный классы.	ПК-2	Доклад, сообщение/ Реферат №3 (10)	Собеседование, опрос/ Контрольная работа №2 (10)	Деловая и (или) ролевая игра/Кейс-задача №1 (20)

09.03.03 Прикладная информатика, направленность (профиль) "Прикладная информатика в экономике"  
 Рабочая программа дисциплины  
 Дисциплина: Б1.В.05 Объектно-ориентированное программирование  
 Форма обучения: очная, заочная  
 Разработана для приема 2021/2022, 2022/2023 учебного года  
 Обновлено на 2023/2024 учебный год

№ п/п	Наименование темы дисциплины	Компетенции	Оценочные средства текущего контроля		
			ЗНАТЬ	УМЕТЬ	ВЛАДЕТЬ
			ПК-2.1	ПК-2.2	ПК-2.3
4	Потоки и файлы.	ПК-2	Доклад, сообщение/ Реферат №4 (10)	Собеседование, опрос/ Контрольная работа №2 (10)	Задания творческого уровня №1 (20)
<b>Количество баллов (100 баллов):</b>			100		

Содержание учебного материала, лабораторные работы и практические занятия, курсовая работа
<p><b>Тема 1:</b> Объектно-ориентированный подход в программировании.                      Сущность объектно-ориентированного подхода в программировании. Цикл разработки программного обеспечения (ПО), назначение и содержание этапов. Роль анализа в процессе разработки программного обеспечения. Основные понятия объектно-ориентированного анализа. Язык C++. Язык Java.  <b>Практические занятия/самостоятельная работа:</b>                      Основы программирования (функции, передача параметров). Создание простейших классов и объектов.  <b>Лабораторная работа:</b> -</p>
<p><b>Тема 2:</b> Классы и объекты.                      Отношения, основные типы отношений. Язык UML. Основные средства анализа и моделирования предметной области в языке UML. Статические данные. Конструктор, деструктор. Операции new и delete. Технология применения объектно-ориентированных языков, их классификация и архитектура. Перегрузка операций. Преобразование типов.  <b>Практические занятия/самостоятельная работа:</b>                      Массивы объектов, указатели и ссылки на объекты (работа с динамическими массивами объектов).  <b>Лабораторная работа:</b> -</p>
<p><b>Тема 3:</b> Наследование, базовый и производный классы.                      Простое и сложное наследование. Абстракция данных, наследование и полиморфизм. Виртуальные функции. Дружественные функции. Дружественные классы. Шаблоны функций. Шаблоны классов. Исключения. Стандартная библиотека шаблонов.  <b>Практические занятия/самостоятельная работа:</b>                      Реализация в коде принципов абстракции данных, наследование и полиморфизм.  <b>Лабораторная работа:</b> -</p>
<p><b>Тема 4:</b> Потоки и файлы.                      Потоки и файлы. Стандартная библиотека классов для управления потоками. Методы и средства организации и программирования интерфейса. Стандарты кодирования и их проекция на объектно-ориентированную модель программирования. Объектный подход к разработке ПО для распределенных систем.  <b>Практические занятия/самостоятельная работа:</b>                      Основные библиотеки много файловых программ и работа с ними.  <b>Лабораторная работа:</b> -</p>
<p><b>Курсовая работа:</b>                      не предусмотрено учебным планом</p>

#### Очная форма обучения

Вид учебной работы	Всего часов	Семестр 5
Аудиторные занятия (АЗ):	54	54
Лекционные занятия (Лек)	18	18
Лабораторные занятия (Лаб)	0	0
Практические занятия (Пр)	36	36
Самостоятельная работа студента (СР)	49	49
Курсовая работа	0	0
Другие виды самостоятельной работы*	49	49
Контроль самостоятельной работы (КСР)	5	5
Контактная работа (КоР)	59	59
Форма промежуточной аттестации	0	Экзамен
Подготовка к экзамену и сдача экзамена (СР, КоР)	36	36
Общая трудоемкость дисциплины, часы/ЗЕТ	144/4	144/4

\* Подготовка к аудиторным занятиям, подготовка к зачету (при наличии)

09.03.03 Прикладная информатика, направленность (профиль) "Прикладная информатика в экономике"  
 Рабочая программа дисциплины  
 Дисциплина: Б1.В.05 Объектно-ориентированное программирование  
 Форма обучения: очная, заочная  
 Разработана для приема 2021/2022, 2022/2023 учебного года  
 Обновлено на 2023/2024 учебный год

№	Наименование темы дисциплины	Семестр/ Курс	Количество учебных часов				Практическая подготовка
			В том числе по видам аудиторных занятий			СР	
			Лек	Пр	Лаб		
1	Объектно-ориентированный подход в программировании.	5	4	8	0	14	8
2	Классы и объекты.	5	6	10	0	13	10
3	Наследование, базовый и производный классы.	5	4	10	0	12	10
4	Потоки и файлы.	5	4	8	0	10	8
Итого:			18	36	0	49	36

\* Практическая подготовка при реализации дисциплин организована путем проведения практических занятий и (или) выполнения лабораторных и (или) курсовых работ и предусматривает выполнение работ, связанных с будущей профессиональной деятельностью.

#### *Заочная форма обучения*

Вид учебной работы	Всего часов	Семестр 6
Аудиторные занятия (АЗ):	10	10
Лекционные занятия (Лек)	4	4
Лабораторные занятия (Лаб)	0	0
Практические занятия (Пр)	6	6
Самостоятельная работа студента (СР)	119	119
Курсовая работа	0	0
Другие виды самостоятельной работы*	119	119
Контроль самостоятельной работы (КСР)	6	6
Контактная работа (КоР)	16	16
Форма промежуточной аттестации	0	Экзамен
Подготовка к экзамену/зачету и сдача экзамена/зачета (СР, КоР)	9	9
Общая трудоемкость дисциплины, часы/ЗЕТ	144/4	144/4

\* Подготовка к аудиторным занятиям

№	Наименование темы дисциплины	Семестр/ Курс	Количество учебных часов				Практическая подготовка
			В том числе по видам аудиторных занятий			СР	
			Лек	Пр	Лаб		
1	Объектно-ориентированный подход в программировании.	6	2	0	0	29	8
2	Классы и объекты.	6	0	2	0	30	10
3	Наследование, базовый и производный классы.	6	2	2	0	30	10
4	Потоки и файлы.	6	0	2	0	30	8
Итого:			4	6	0	119	36

\* Практическая подготовка при реализации дисциплин организована путем проведения практических занятий и (или) выполнения лабораторных и (или) курсовых работ и (или) путем выделения часов из часов, отведенных на самостоятельную работу, и предусматривает выполнение работ, связанных с будущей профессиональной деятельностью.

#### 4. Способ реализации дисциплины

Без использования онлайн-курса.

#### 5. Учебно-методическое обеспечение дисциплины:

*Основная литература:*

1. ПРОГРАММИРОВАНИЕ. ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД. Учебник и практикум для вузов / Зыков С. В. - Национальный исследовательский университет «Высшая школа экономики» (г. Москва), 2022 г. - 155 с. - ISBN 978-5-534-00850-0 – Режим доступа: <https://urait.ru/book/programmirovanie-obektno-orientirovannyu-podhod-490423>

2. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ. Учебное пособие для

вузов / Тузовский А. Ф. - Национальный исследовательский Томский политехнический университет (г. Томск), 2022 г. - 206 с. - ISBN 978-5-534-00849-4 – Режим доступа: <https://urait.ru/book/obektno-orientirovannoe-programmirovanie-490369>

3. ПРОГРАММНАЯ ИНЖЕНЕРИЯ. ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ ПРОГРАММНЫХ СИСТЕМ 2-е изд., испр. и доп. Учебник для вузов / Черткова Е. А. - Национальный исследовательский университет «Высшая школа экономики» (г. Москва), 2022 г. - 147 с. - ISBN 978-5-534-09172-4 – Режим доступа: <https://urait.ru/book/programmnyaya-inzheneriya-vizualnoe-modelirovanie-programmnyh-sistem-491629>

*Дополнительная литература:*

1. Рацеев С. М. — Программирование на языке Си. - Издательство Лань, 2022 г. - 332 с. - ISBN 978-5-8114-8585-7 – Режим доступа: <https://e.lanbook.com/book/193320>

3. ОСНОВЫ ПРОГРАММИРОВАНИЯ. Учебник и практикум для вузов / Черпаков И. В. - Финансовый университет при Правительстве РФ (г. Москва), 2022 г. - 219 с. - ISBN 978-5-9916-9983-9 – Режим доступа: <https://urait.ru/book/osnovy-programmirovaniya-489747>

3. ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ. Учебное пособие для вузов / Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю. - Санкт-Петербургский государственный экономический университет (г. Санкт-Петербург), 2022 г. - 235 с. - ISBN 978-5-534-02816-4 – Режим доступа: <https://urait.ru/book/tehnologii-i-metody-programmirovaniya-489920>

**6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения**

1. Операционная система
2. Пакет прикладных офисных программ
3. Антивирусное программное обеспечение
4. LMS Moodle
5. Вебинарная платформа
6. Java SE
7. Visual Studio Code

**7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», информационных справочных систем и профессиональных баз данных, необходимых для освоения дисциплины**

1. [ibooks.ru](https://ibooks.ru) : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://ibooks.ru>. - Текст: электронный

2. Электронно-библиотечная система СПБУТУиЭ : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <http://libume.ru>. - Текст: электронный

3. Юрайт : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://urait.ru>. - Текст: электронный

4. [eLibrary.ru](http://elibrary.ru) : научная электронная библиотека [Электронный ресурс] : профессиональная база данных. - Режим доступа: <http://elibrary.ru>. - Текст: электронный

5. Архив научных журналов НЭИКОН [Электронный ресурс] : профессиональная база данных. - Режим доступа: [arch.neicon.ru](http://arch.neicon.ru). - Текст: электронный

6. КиберЛенинка : научная электронная библиотека [Электронный ресурс] : информационная справочная система. - Режим доступа: <http://cyberleninka.ru>. - Текст: электронный

7. Лань : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://e.lanbook.com>. - Текст: электронный

8. [it-world.ru](https://www.it-world.ru) [Электронный ресурс] : информационная справочная система . - Режим доступа: <https://www.it-world.ru>. - Текст: электронный

9. Connect: IT-технологии : информационная справочная система. - Режим доступа: <https://www.connect-wit.ru/>. - Текст: электронный

10. Компьютерра : информационная справочная система . - Режим доступа: <https://www.computerra.ru/>. - Текст: электронный

11. Виртуальный компьютерный музей [Электронный ресурс] : информационная справочная система . - Режим доступа: <https://www.computer-museum.ru>. - Текст: электронный

12. Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации: профессиональная база данных. - Режим доступа: <https://digital.gov.ru>. - Текст: электронный

13. Федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций: профессиональная база данных . - Режим доступа: <https://rkn.gov.ru>. - Текст: электронный

14. Math-Net.Ru: профессиональная база данных . - Режим доступа: <https://www.mathnet.ru/>. - Текст: электронный

## **8. Материально-техническое обеспечение дисциплины**

1. Учебная аудитория для проведения занятий лекционного типа, семинарского типа - практических занятий, для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оборудованная: рабочими местами для обучающихся, оснащенные специальной мебелью; рабочим местом преподавателя, оснащенный специальной мебелью, персональным компьютером с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; техническими средствами обучения - мультимедийным оборудованием (проектор, экран, колонки) и маркерной доской; лицензионным программным обеспечением

2. Учебная аудитория для проведения занятий семинарского типа - практических занятий – компьютерный класс, оборудованный рабочими местами для обучающихся, оснащенные специальной мебелью, персональными компьютерами с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; рабочим местом преподавателя, оснащенный специальной мебелью, персональным компьютером с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; техническими средствами обучения - мультимедийным оборудованием (проектор, экран, колонки) и маркерной доской; лицензионным программным обеспечением

3. При применении электронного обучения, дистанционных образовательных технологий используются: виртуальные аналоги учебных аудиторий - вебинарные комнаты на вебинарных платформах, рабочее место преподавателя, оснащенное персональным компьютером (планшет, мобильное устройство) с возможностью подключения к сети «Интернет», доступом к электронной информационно-образовательной среде Университета и к информационно-образовательному portalу Университета [imeos.ru](http://imeos.ru), веб-камерой, микрофоном и гарнитурой (в т.ч. интегрированными в устройства), программным обеспечением; рабочее место обучающегося оснащено персональным компьютером (планшет, мобильное устройство) с возможностью подключения к сети «Интернет», доступом к электронной информационно-образовательной среде Университета и к информационно-образовательному portalу Университета [imeos.ru](http://imeos.ru), веб-камерой, микрофоном и гарнитурой (в т.ч. интегрированными в устройства). Авторизация на информационно-образовательном portalе Университета [imeos.ru](http://imeos.ru) и начало работы осуществляются с использованием персональной учетной записи (логина и пароля). Лицензионное программное обеспечение

4. Помещение для самостоятельной работы, оборудованное специальной мебелью, персональными компьютерами с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета, лицензионным программным обеспечением

## 9. Оценочные материалы по дисциплине

Описание оценочных средств (показатели и критерии оценивания, шкалы оценивания) представлено в приложении к основной профессиональной образовательной программе «Каталог оценочных средств текущего контроля и промежуточной аттестации».

Процедуры оценивания знаний, умений, навыков и (или) опыта деятельности приводятся в соответствующих методических материалах и локальных нормативных актах Университета.

Для оценивания учебных достижений студентов в Университете действует балльно-рейтинговая система.

Если оценка, соответствующая набранной в семестре сумме рейтинговых баллов, удовлетворяет студента, то она является итоговой оценкой по дисциплине при проведении промежуточной аттестации в форме экзамена/зачета с оценкой/зачета.

Условием сдачи экзамена/зачета с оценкой/зачета с целью повышения итоговой оценки по дисциплине является сдача студентом экзамена, за который он получает экзаменационные баллы без учета баллов, полученных за текущий контроль:

### Шкала оценивания учебных достижений по дисциплине, завершающейся зачетом без оценки

Баллы по дисциплине	60 и менее		61-73		74-90		91-100
Итоговая оценка по дисциплине	Незачет		Зачет				
Баллы в международной шкале ECTS с буквенным обозначением уровня	50 и менее	51-60	61-67	68-73	74-83	84-90	91-100
	F	Fx	E	D	C	B	A
Уровень сформированности компетенций	Не сформированы		Пороговый		Высокий		Повышенный

### Шкала оценивания учебных достижений по дисциплине, завершающейся экзаменом/зачетом с оценкой

Баллы по дисциплине	60 и менее		61-73		74-90		91-100
Итоговая оценка по дисциплине	Неудовлетворительно		Удовлетворительно		Хорошо		Отлично
Баллы в международной шкале ECTS с буквенным обозначением уровня	<50	51-60	61-67	68-73	74-83	84-90	91-100
	F	Fx	E	D	C	B	A
Уровень сформированности компетенций	Не сформированы		Пороговый		Высокий		Повышенный

## 9.1. Типовые контрольные задания для текущего контроля

### Доклад, сообщение / Реферат № 1

1. Что будет, если единственный конструктор класса объявлен как final?
2. Что такое finalize? Зачем он нужен? Что Вы можете рассказать о сборщике мусора и алгоритмах его работы.
3. Почему метод clone объявлен как protected? Что необходимо для реализации клонирования?
4. Есть ли какие-либо рекомендации о том, какие поля следует использовать при подсчете hashCode?
5. Как вы думаете, будут ли какие-то проблемы, если у объекта, который используется в качестве ключа в hashMap изменится поле, которое участвует в определении hashCode?
6. Чем отличается абстрактный класс от интерфейса, в каких случаях что вы будете использовать?

### Доклад, сообщение / Реферат № 2



1. Расскажите про каждый из методов класса Object.
2. Что такое метод equals(). Чем он отличается от операции ==.
3. Если вы хотите переопределить equals(), какие условия должны удовлетворяться для переопределенного метода?
4. Если equals() переопределен, есть ли какие-либо другие методы, которые следует переопределить?

### **Доклад, сообщение / Реферат № 3**

1. Объекты синхронизации пакета java.util.concurrent
2. Объект синхронизации Semaphore
3. Объект синхронизации CountdownLatch
4. Объект синхронизации CyclicBarrier
5. Объект синхронизации Exchanger
6. Объект синхронизации Phaser
7. Потокбезопасные коллекции
8. Атомарные классы пакета util.concurrent
9. Блокирующие и неблокирующие очереди
10. Блокировки пакета concurrent

### **Доклад, сообщение / Реферат № 4**

1. Синхронизация потоков
2. Поток-демон
3. Приостановка потока
4. Использование метода join
5. Взаимная блокировка потоков
6. Методы interrupt, interrupted, isInterrupted
7. «Гонка» в многопоточном приложении
8. «Голодание» в многопоточном приложении

### **Собеседование, опрос / Контрольная работа №1**

Задача 1. Для последовательности Фибоначчи найти:

- а) n-й элемент последовательности по данному номеру n;
- б) первое число Фибоначчи, превосходящее данное M, а также номер этого числа в последовательности;
- в) сумму всех элементов с номера n по номер m и вывести их на экран

Задача 2. Сумма цифр трехзначного числа кратна 7. Само число также делится на 7. Найти все такие числа

Задача 3. Квадрат трехзначного числа оканчивается тремя цифрами, которые как раз составляют взятое число. Найти все такие числа

### **Собеседование, опрос / Контрольная работа №2**

Задание 1. Ответьте на следующие вопросы:

- а) переменная, которая известна только внутри функции, в которой она определена, называется \_\_\_\_\_
- б) оператор \_\_\_\_\_ используется для передачи значения вызывающей функции
- в) ключевое слово \_\_\_\_\_ используется в заголовке функции для того чтобы показать, что функция не возвращает значения или не принимает никаких параметров
- г) \_\_\_\_\_ позволяет компилятору проверять количество, тип и порядок

---

следования аргументов, переданных функции

- д) спецификаторами класса памяти являются \_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_ и \_\_\_\_\_
- е) предполагается, что переменные, объявленные в теле функции или списке параметров функции, имеют класс памяти \_\_\_\_\_, если не указано иначе
- ж) спецификатор \_\_\_\_\_ рекомендует компилятору разместить переменную в одном из регистров компьютера
- з) чтобы локальная переменная сохраняла свое значение между вызовами функций, она должна быть объявлена со спецификатором \_\_\_\_\_
- к) функция, которая вызывает сама себя, называется \_\_\_\_\_ функцией

Задание 2. Написать функцию, определяющую, имеют ли два вещественных числа двойной точности одинаковые знаки.

Задание 3. Напишите программу, выполняющую следующие действия с массивом  $n$  целых элементов: ввод значений с клавиатуры, вывод на экран содержимого массива, нахождение максимального элемента массива, нахождение минимального элемента массива, умножение всех элементов массива на число. Работу пользователя с массивом организовать с помощью меню. Каждое действие с массивом оформить в виде функции. Работу с функциями организовать с помощью массива указателей на функции.

#### **Деловая и (или) ролевая игра / Кейс-задача №1**

Задание 1. Создать структуру с именем `time`. Три её поля, имеющие тип `int`, будут называться `hours`, `minutes` и `seconds`. Написать программу, которая просит пользователя ввести время в формате часы, минуты, секунды. Программа должна хранить время в структурной переменной типа `time` и выводить количество секунд в введенном времени.

Задание 2. Создать перечисление с именем `pets` и значениями `dog`, `cat`, `rat`, `fish`, `bird`. Создать структуру с именем `animal`. Определить её поля: `name` как массив из 20 символов типа `char`, `type` как типа `pets` и `age` типа `float`, хранящие, соответственно, имя животного, его разновидность и возраст. Написать программу, которая создает 4 переменные типа `animal` и просит пользователя ввести значения разновидности, имени и возраста всех 4 животных. Причем разновидность животного вводится по первому символу, набранному на клавиатуре (т.е. если нажата `d`, то в соответствующее поле вводится значение `dog`). В качестве результата программа должна выдать список всех имеющихся животных со всеми значениями полей

Задание 3. Создать класс с именем `Time`, содержащий три поля типа `int`, предназначенных для хранения часов, минут и секунд. Один из конструкторов класса должен инициализировать поля нулевыми значениями, а другой – заданным набором значений. Создать метод класса, который будет выводить значения полей на экран, в формате `23:59:59` и метод, складывающий значения двух объектов типа `Time`, передаваемых ему в качестве аргументов. Продемонстрировать работу класса.

Задание 4. Создать класс с именем `fraction`, содержащий два поля типа `int` – числитель и знаменатель обыкновенной дроби. Конструктор класса должен инициализировать их заданным набором значений. Создать метод класса, который будет выводить дробь на экран в формате  $x/y$ , метод, складывающий две дроби, переданные ему в параметрах и метод, умножающий две дроби, переданные ему в параметрах.

Задание 5. Описать класс `fraction`, у которого поля  $x$  и  $y$  задают числитель и знаменатель обыкновенной дроби. Перегрузить для этого класса арифметические операции сложения, вычитания, умножения и деления так, чтобы они могли оперировать как с объектами класса, так и с числами (то есть выполнять, например, не только действие  $3/4 + 2/5$ , но и  $1/2 + 4$  или  $2 * 5/6$ ). Также перегрузить операции сравнения `==` и `!=`. Продемонстрировать работу класса.

### **Задание творческого уровня №1**

Напишите программу, разработайте систему тестов для верификации программы, подготовьте отчет о степени соответствия программы требованиям.

**Задание 1.** Определить класс `vector2` как вектор на плоскости с данными `x` и `y`. Определить для него операции сложения, вычитания и скалярного произведения, определить функцию присваивания значения координатам вектора и функцию вывода значений на консоль. Определить класс `vector3` как вектор в пространстве, породив его от класса `vector2`. Переопределить для него функции и операции. Продемонстрировать работу класса.

**Задание 2.** Создать абстрактный класс `Figure` с виртуальными методами вычисления площади и периметра. Создать производные классы: `Rectangle` (прямоугольник), `Circle` (круг), `Triangle` (треугольник). Описать в производных классах функции вычисления периметра и площади, продемонстрировать работу механизма виртуальных функций.

**Задание 3.** Создать шаблон функции, возвращающей среднее арифметическое всех элементов массива. Аргументами функции должны быть имя и размер массива (типа `int`). Создать шаблон функции, возвращающей значение максимального элемента массива. Аргументами функции должны быть имя и размер массива (типа `int`). Создать шаблон функции, обменивающей местами значения двух передаваемых ей по ссылке аргументов. Создать шаблон функции, осуществляющей сортировку данных массива. Аргументами функции должны быть имя и размер массива (типа `int`). Продемонстрировать работу шаблонов на данных различных типов, в том числе, на данных вновь созданного класса – вектор на плоскости, определив для него операции сравнения.

### **9.2. Примерный перечень тем курсовой работы**

Не предусмотрено учебным планом

### **9.3. Типовые контрольные задания для промежуточной аттестации: экзамен**

#### **Примерный перечень теоретических вопросов к экзамену: Вопрос № 1**

№	Задание	Варианты ответа	Кол-во баллов
1.	Объектно-ориентированное программирование характеризуется:	Наличием одной линейной программы Разделением программы на модули все данные об объекте, его связи с другими объектами объединяются в одну структурную переменную	4
2.	Метод это:	Функция, получая в качестве обязательного параметра указатель на объект Структура, хранящая указатели <code>this</code> , <code>parent</code> , <code>super</code> Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии Определенный программистом абстрактный тип данных	4
3	Объект это:	Функция, получая в качестве обязательного параметра указатель на объект Структура, хранящая указатели <code>this</code> , <code>parent</code> , <code>super</code> Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии Определенный программистом абстрактный тип данных	4

4	Класс это:	Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии Определенный программистом абстрактный тип данных Переменная, описанная абстрактным типом данных	4
5	Совокупность формальных параметров определяет	тип возвращаемого функцией значения; сигнатуру функции; идентификатор функции; последовательность описаний и определений функции.	4
6	Какое из свойств скрывает внутренние данные объекта?	Инкапсуляция Полиморфизм Наследование Объектно-Ориентированные	4
7	Какая из команд описывает переменную, которая не является файловой?	F: File; F: TextFile; F: File of real; F: FileText;	4
8	В программе описано класс и объект class A {public: int a, b, c; }; A * obj; Как обратиться к атрибуту c?	obj.c obj-> c obj A -> -> c obj-> A.c	4
9	Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам?	Шаблонная. Полиморфная. Дружеская. Статическая.	4
10	В каких случаях надо иметь в классе конструктор копирования:	Когда нужно передать элементы класса Для выполнения операции присвоение одного элемента другому Для создания дружественной функции Для уничтожения объекта из памяти	4
11	Какое из свойств строит иерархию объектов?	Инкапсуляция Полиморфизм Наследование Объектно-Ориентированные	4
12	Цикл с постусловием определяется служебным словом	FOR WHILE CASE REPEAT	4
13	Каких параметров подпрограммы не существует	параметров-переменных параметров-констант параметров-значений параметров-модулей	4
14	Какие виды подпрограмм есть в паскале	модуль и функция процедура и функция модуль и процедура процедура	4
15	Какой из параметров записан правильно?	var y = array [1..10] of integer. var y : array (1..10) of integer; var y : array [1..10] of integer;+ type y = array (1..10) of integer	4

**Примерный перечень практических заданий к экзамену: Вопрос № 2**

1	Выполните код и определите, какую функцию он выполняет <pre>{x,max=0;          { printf("x="); scanf("%d",&amp;x);(x&gt;=-100 &amp;&amp; x&lt;=100 &amp;&amp; max=-100 &amp;&amp; x&lt;=100);("max=%d",max); getch(); }</pre>	Ищет наименьшее число Ищет наибольшее число Ищет наибольшее число по модулю Сортирует массив	10
2	Выполните код и определите, какую функцию он выполняет <pre>{                                const n=7;i,ch=0,nch=0;a[n];(i=1;i&lt;=n;i++) { printf("a[%d]=",i); scanf("%d",&amp;a[i]);(a[i]%2==0) ch++; else nch++; }("ch=%d  nch=%d",ch,nch); getch(); }</pre>	Ищет в массиве количество четных и нечетных элементов Ищет в массиве количество четных элементов Ищет в массиве количество нечетных элементов Ищет в массиве элементы, равные нулю	10
3	Дан объект данных, над которым определены операции согласно вариантам. Реализовать набор операций для работы с объектом так, чтобы его можно было использовать в выражениях, не прибегая к вызову функций. Необходимо разработать: - класс объекта и определить правила выполнения операций над ним; - набор перегруженных операторов, реализующих операции с объектом; - интерфейс для редактирования объекта с помощью операторов; - интерфейс для тестирования использования объекта в выражениях. Объект: интервал времени (часы, минуты, секунды). Реализовать операции с учётом ограничений на часы (0 до 23), минуты и секунды (0 до 59), т.е. результат всегда от 0:0:0 до 23:59:59. Принять: (+ и +=) – сложение, (- и -=) – вычитание, (* и *=) – удлинение или сокращение, (!) – проверка на ноль, (== и !=) – сравнение, (long) – преобразование в секунды, (float) – преобразование в часы (3600 сек), (~) – дополнение до конца суток.		20

Раздел билета	Компетенции	Планируемые результаты обучения по дисциплине	Количество баллов
Вопрос №1 Теоретический вопрос (проверяет знания («знать»), сформированные дисциплиной)	ПК-2	Знать основы объектно-ориентированного программирования.	60
Вопрос №2 Практическое задание (проверяет умения («уметь»), проверяет практические навыки («владеть»), сформированные дисциплиной)	ПК-2	Уметь алгоритмизировать деятельность; кодировать на языках объектно-ориентированного программирования; тестировать результаты прототипирования. Владеть навыками проведения анализа результатов тестов; координирования и проведения оценки готовых систем; обработки и анализа результатов оценки готовых систем на соответствие требованиям.	40