**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КРЫМ**

**«КРЫМСКИЙ ИНЖЕНЕРНО-ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»**

**Кафедра прикладной информатики**

|  |  |
| --- | --- |
| «СОГЛАСОВАНО»  Руководитель ОПОП  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ильясова Ф.С.  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_2017 года | «УТВЕРЖДАЮ»  Заведующий кафедрой  \_\_\_\_\_\_\_\_\_\_\_\_ Сейдаметова З.С.  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_2017 года |

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**«Б1.В.ОД.6 РАЗРАБОТКА СЕРВЕРНЫХ ПРИЛОЖЕНИЙ»**

Направление подготовки

**09.03.03 Прикладная информатика**

Профиль

**Прикладная информатика**

Факультет экономики, менеджмента и информационных технологий

Симферополь, 2017

Рабочая программа дисциплины «Разработка серверных приложений» для бакалавров направления подготовки 09.03.03 Прикладная информатика профиля «Прикладная информатика» составлена на основании ФГОС ВО, утвержденного приказом Министерства образования и науки РФ от 12.03.2015 г. № 207, и учебного плана по направлению подготовки 09.03.03 Прикладная информатика, утвержденного Ученым советом ГБОУВО РК КИПУ от 24.04.17г., протокол № 12

Составитель рабочей программы к.т.н., доц. Абдурайимов Л.Н.

Рабочая учебная программа утверждена на кафедре прикладной информатики

Протокол № 1 от 30 августа 2017 г.

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сейдаметова З.С.

Рабочая программа одобрена на заседании УМК факультета экономики, менеджмента и информационных технологий

Протокол № \_\_\_\_ от «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2017 г.

Председатель УМК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Рабочая учебная программа переутверждена на заседании кафедры прикладной информатики

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

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сейдаметова З.С.

***Примечание****: РПД должны утверждаться датой, предшествующей дате утверждения Ученым советом университета ОПОП по направлению подготовки.*

**Содержание рабочей программы и методических материалов к РПД**

[1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы 5](#_Toc467780739)

[*Цель и задачи изучения дисциплины* 5](#_Toc467780740)

[*Ожидаемые результаты освоения дисциплины* 5](#_Toc467780741)

[2. Место дисциплины в структуре образовательной программы 6](#_Toc467780742)

[3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу 7](#_Toc467780743)

[4. Содержание дисциплины (структурированное по темам с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий) 7](#_Toc467780744)

[4.1. Содержание дисциплины, структурированное по темам 7](#_Toc467780745)

[4.2. Тематический план лекций 10](#_Toc467780746)

[4.3. Темы лабораторных работ 10](#_Toc467780747)

[5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине 10](#_Toc467780748)

[5.1. Содержание самостоятельной работы студентов по дисциплине 10](#_Toc467780749)

[6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине 11](#_Toc467780750)

[6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы 11](#_Toc467780751)

[6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания 12](#_Toc467780752)

[6.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы 13](#_Toc467780753)

[6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций 15](#_Toc467780754)

[6.5. Итоговая рейтинговая оценка текущей и промежуточной аттестации студента по дисциплине 16](#_Toc467780755)

[7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины 17](#_Toc467780756)

[8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины 18](#_Toc467780757)

[9. Методические указания для обучающихся по освоению дисциплины 19](#_Toc467780758)

[9.1. Общие рекомендации по самостоятельной работе студентов 19](#_Toc467780759)

[9.2. Методические рекомендации по освоению лекционного материала, подготовке к лекциям 20](#_Toc467780760)

[9.3. Методические рекомендации по подготовке к практическим занятиям 21](#_Toc467780761)

[10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (включая перечень программного обеспечения и информационных справочных систем (при необходимости)) 22](#_Toc467780762)

[11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине 23](#_Toc467780763)

[Оформление комплекта заданий для контрольной работы 24](#_Toc467780764)

**Рабочая программа дисциплины «Разработка серверных приложений»**

для бакалавров направления подготовки 09.03.03 Прикладная информатика профиля «Прикладная информатика»

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

## *Цель и задачи изучения дисциплины*

*Цель дисциплины* – освоение студентами навыков решения практических задач с использованием языка программирования высокого уровня Java, освоение технологии объектно-ориентированного программирования, использование различных структур и алгоритмов обработки данных, методов программирования и реализации графического пользовательского интерфейса.

*Задачи дисциплины* заключаются в следующем:

* изучение языка программирования и платформы Java;
* изучение базовых понятий и принципов объектно-ориентированного программирования, а также технических аспектов и методологии объектно-ориентированного программирования;
* углубленное изучение методов и инструментальных средств объектно- ориентированного программирования;
* знакомство с библиотеками классов, широко используемых при создании прикладных программ;
* формирование социально-личностных качеств студентов: целеустремленности, организованности, трудолюбия, коммуникативности, готовности к деятельности в профессиональной среде, ответственности за принятие профессиональных решений.

Изучение методологии объектно-ориентированного программирования предусматривает изложение концепций объектно-ориентированного программирования и средств языка программирования Java для их реализации. В первую очередь рассматривается инструментальная среда разработки программ, что является необходимым для продуктивного выполнения лабораторных работ.

## *Ожидаемые результаты освоения дисциплины*

В результате освоения дисциплины должны быть сформированы следующие компетенции:

**Профессиональные компетенции (ПК):**

* способностью проводить научные эксперименты, оценивать результаты исследований (**ПК-4**);
* способностью проектировать архитектуру и сервисы ИС предприятий и организаций в прикладной области (**ПК-12**).

Сформированность указанных компетенций определяется тем, что студент должен:

**знать:** основные концепции объектно-ориентированного программирования (инкапсуляция, наследования и полиморфизм), основные конструкции языка программирования Java, основные принципы организации сложных объектно-ориентированных систем, средства объектно-ориентированного программирования на Java, методику объектно-ориентированного анализа и проектирования;

**уметь:** писать программы на языке Java с использованием объектно-ориентированного подхода, применять приемы и методы ООП в своей практической деятельности, самостоятельно решать поставленные задачи;

**владеть:** представлением об основных тенденциях развития современных информационных технологий и объектно-ориентированных библиотеках, информацией о технических аспектах реализации объектно-ориентированных принципах в языках программирования.

Результаты освоения дисциплины «Разработка серверных приложений» достигаются за счет использования в процессе обучения различных, в том числе интерактивных, методов и технологий формирования указанных компетенций.

Предусматриваются следующие формы организации учебных занятий: лекции с проблемной постановкой темы; интерактивное обсуждение тем, подготовленных студентами самостоятельно; лабораторные работы, выполнение индивидуальных заданий по углубленному изучению отдельных компонентов современных встроенных систем; электронное тестирование знаний.

# 2. Место дисциплины в структуре образовательной программы

Дисциплина «Разработка серверных приложений» относится к обязательным дисциплинам вариативной части учебного плана подготовки бакалавров направления подготовки 09.03.03 Прикладная информатика. Дисциплина базируется на знаниях, умениях и навыках, приобретенных в результате изучения дисциплин подготовки бакалавров направления подготовки 09.03.03 Прикладная информатика:

* Информатика и программирование;
* Базы данных;
* Алгоритмы и структуры данных;
* Объектно-ориентированное программирование;
* Операционные системы.

Учебные курсы, для которых необходимы знания, умения, навыки, приобретаемые в результате изучения дисциплины «Разработка серверных приложений»:

* Системное программирование;
* Параллельные и распределенные вычисления;
* Программирование и поддержка веб-приложений;
* Программная инженерия.

Знания, навыки и умения, приобретенные в результате прохождения курса, будут востребованы при изучении дисциплин специализаций, связанных с созданием информационных систем, веб-программированием, выполнением курсовых и дипломных работ, работой над задачами во время прохождения производственной практики.

# 3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Семестр | Общее количество часов | Количество зачетных единиц | Контактные часы | | | | | СР | контроль | Итоговый контроль (экзамен, зачет) |
| Всего | л | п | л/р | КСР |
| ОФО | | | | | | | | | | |
| 5 | 144 | 4,0 | 68 | 30 | – | 34 | 4 | 49 | 27 | экзамен |
| ЗФО | | | | | | | | | | |
| 6 | 144 | 4,0 | 16 | 6 | – | 8 | 2 | 119 | 9 | экзамен |

*Сокращения: Л – лекции П – практические занятия*

*Л/р – лабораторные занятия СР – самостоятельная работа*

*ОФО – очная форма обучения ЗФО – заочная форма обучения*

# 4. Содержание дисциплины (структурированное по темам с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий)

## 4.1. Содержание дисциплины, структурированное по темам

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Наименования тем  (разделов, модулей) | Количество часов | | | | | | | | | | | | Формы текущего контроля |
| очная форма | | | | | | заочная форма | | | | | |
| всего | в том числе | | | | | всего | в том числе | | | | |
| л | п | с | л/р | с/р | л | п | с | л/р | с/р |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Раздел 1. Введение в Разработка серверных приложений. Синтаксис языка. | | | | | | | | | | | | | |
| Тема 1. Первая Java программа Среда программирования Eclipse IDE. | 5 | 2 | – | – | – | 3 | 6 | 1 | – | – | – | 5 | 1. Тестовый контроль в системе управления курсами Moodle.  2. Отчеты по лабораторным работам. |
| Тема 2. Базовый синтаксис Java. | 18 | 4 | – | – | 8 | 6 | 15 | 1 | – | – | 2 | 12 |
| Тема 4. Кусочки: массивы, циклы, пакеты, уровни доступа, аргументы командной строки. | 12 | 4 | – | – | 4 | 4 | 13 | 1 | – | – | 2 | 10 |
| Раздел 2. Введение в объектно-ориентированное программирование. | | | | | | | | | | | | | |
| Тема 3. Введение в Объектно-ориентированное программирование. Классы, методы, свойства. | 14 | 4 | – | – | 6 | 6 | 13 | 1 | – | – | – | 12 | 1. Тестовый контроль в системе управления курсами Moodle.  2. Отчеты по лабораторным работам. |
| Тема 5. Кастинг, абстрактные классы, интерфейсы, полиморфизм. | 12 | 4 | – | – | 4 | 6 | 17 | 1 | – | – | 2 | 14 |
| Раздел 3. Графические интерфейсы пользователя. | | | | | | | | | | | | | |
| Тема 6. Введение в GUI с использованием Swing. Событийно-управляемое программирование. | 10 | 2 | – | – | 4 | 4 | 15 | 1 | – | – | – | 14 | 1. Тестовый контроль в системе управления курсами Moodle.  2. Отчеты по лабораторным работам. |
| Тема 7. Введение в GUI с использованием Swing (часть 2). Внутренние классы. Введение в HTML и Апплеты. Установка сервера Apache Tomcat. | 10 | 2 | – | – | 4 | 4 | 10 | – | – | – | – | 10 |
| Раздел 4. Исключения. Java коллекции. Генерики. | | | | | | | | | | | | | |
| Тема 8. Обработка ошибок. Исключения. | 6 | 2 | – | – | – | 4 | 10 | – | – | – | – | 10 | 1. Тестовый контроль в системе управления курсами Moodle.  2. Отчеты по лабораторным работам. |
| Тема 9. Избранные Java коллекции. Generics. | 8 | 2 | – | – | – | 4 | 10 | – | – | – | – | 10 |
| Раздел 5. Сетевое программирование. | | | | | | | | | | | | | |
| Тема 10. Работа с потоками ввода/вывода. Java сериализация. Сетевые основы. | 12 | 2 | – | – | 4 | 4 | 15 | – | – | – | 2 | 12 | 1. Тестовый контроль в системе управления курсами Moodle.  2. Отчеты по лабораторным работам. |
| Тема 11. Получение цен на акции. Сокеты. RMI. | 6 | 2 | – | – | – | 4 | 11 | – | – | – | – | 10 |
| **Всего часов[[1]](#footnote-1)** | **144** | **30** | **–** | **–** | **34** | **49** | **144** | **6** | **–** | **–** | **8** | **119** |  |
| Форма итогового контроля | экзамен  КСР – 4 часа  контроль – 27 часов | | | | | | экзамен  КСР – 2 часа  контроль – 9 часов | | | | | |  |

*\*\*\* сокращения:*

*Л – лекции Л/р – лабораторные занятия С – семинарские занятия*

*П – практические занятия С/р – самостоятельная работа*

## 4.2. Тематический план лекций

Тематический план лекций приведен в п. 4.1. Форма проведения лекционных занятий – с использованием мультимедиа оборудования, электронных презентационных материалов с поддержкой виртуализации обучения, т. е. активное использование системы управления курсами Moodle с авторскими разработками.

## 4.3. Темы лабораторных работ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  занятия | Тема занятия | Форма проведения (актив./ интерактив.) | Количество часов | |
| ОФО | ЗФО |
| 1 | Тема: Введение в классы и объекты. | актив | 4,0 | 1,0 |
| 2 | Тема: Типы данных и операторы. | актив | 4,0 | 1,0 |
| 3 | Тема: Классы и объекты. | актив | 4,0 | 1,0 |
| 4 | Тема: Наследование и полиморфизм. | актив | 4,0 | 1,0 |
| 5 | Тема: Обработка изображений. Пакетная обработка файлов. | актив | 4,0 | 1,0 |
| 6 | Тема: Графические интерфейсы пользователя. | актив | 6,0 | 1,0 |
| 7 | Тема: Разработка веб-сервиса на Java. | актив | 8,0 | 2,0 |
|  | **Итого:** |  | **34,0** | **8,0** |

# 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

1. Блинов И.Н., Романчик В.С. Java. Методы программирования: уч.-мет. пособие. – Минск: Издательство «Четыре четверти», 2013. – 896 с.
2. Шилдт Г. Java 8: руководство для начинающих, 6-е изд.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2015. – 720 с.
3. Арнольд К., Гослинг Д. Разработка серверных приложений. – СПб.: Питер, 1997. – 304 с.
4. Методические рекомендации по выполнению лабораторных работ по дисциплине «Разработка серверных приложений» для студентов очной и заочной форм обучения направления подготовки 09.03.03 «Прикладная информатика» / Л.Н. Абдурайимов. – ГБОУВО РК «КИПУ», 2015-2016.
5. Методические рекомендации к выполнению контрольной работы по дисциплине «Разработка серверных приложений» для студентов заочной формы обучения направления подготовки 09.03.03 «Прикладная информатика» / Л.Н. Абдурайимов. – ГБОУВО РК «КИПУ», 2016-2017.

## 5.1. Содержание самостоятельной работы студентов по дисциплине

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование тем и вопросы, выносимые на самостоятельную работу | Форма отчетности  (формы СРС и вырабатываемые компетенции) | Кол-во часов | |
| ОФО | ЗФО |
| 1 | Тема 1. Введение. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 4,0 | 7,0 |
| 2 | Тема 2. Инструментарий. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 4,0 | 8,0 |
| 3 | Тема 3. Конструкции языка. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 6,0 | 10,0 |
| 4 | Тема 4. Коллекции. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 6,0 | 10,0 |
| 5 | Тема 5. Исключения. Модульное тестирование. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 4,0 | 12,0 |
| 6 | Тема 6. Java Foundation Classes (JFC). | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 4,0 | 10,0 |
| 7 | Тема 7. Ввод и вывод. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 4,0 | 14,0 |
| 8 | Тема 8. Сериализация. Java Beans. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 5,0 | 16,0 |
| 9 | Тема 9. Многопоточные приложения. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 6,0 | 16,0 |
| 10 | Тема 10. Базы данных. | Работа с литературой, подготовка к лабораторным работам, текущему тестовому контролю, выполнение домашней работы | 6,0 | 16,0 |
|  | **Итого:** |  | **49,0** | **119,0** |

*\*\*\* сокращения:*

*ОФО – очная форма обучения*

*ЗФО – заочная форма обучения*

***Примечание:*** *самостоятельная работа студентов включает самостоятельное изучение отдельных тем курса, подготовку к лабораторным, практическим и семинарским занятиям, текущему контролю, составление учебных рефератов и др.*

*Графа 3 может содержать такие формы, как работа с литературой, выполнение письменной домашней работы и т.д.*

# 6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

*Методы контроля и самоконтроля по эффективности учебно-познавательной деятельности:*

*а) устный контроль и самоконтроль: индивидуальные и фронтальные опросы, устные зачеты и экзамены, программированные опросы, устные самоконтроль;*

*б) письменный контроль и самоконтроль: контрольные письменные работы, письменные зачеты и экзамены, программированные письменные работы, письменные самоконтроль;*

*в) лабораторно-практический контроль и самоконтроль: контрольно-лабораторные работы, машинный контроль, лабораторно-практический самоконтроль.*

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

|  |  |  |  |
| --- | --- | --- | --- |
| Дескрип­торы | Компетенции | | Оценочные средства |
| Шифр компетенции ПК-2 | Шифр компетенции ПК-12 |
| 1 | 2 | 3 | 4 |
| Знать | Основы объектно-ориентированного программирования.  Лексику языка, типы данных в Java. | Методы объявления классов, преобразования типов в Java.  Что такое объектная модель в Java. | Аналитический обзор по самостоятельному изучению теоретического материала, тестовый контроль, вопросы к экзамену. |
| Уметь | Реализовывать задачи, используя синтаксис языка Java.  Реализовывать задачи с применением методов вывода в интерактивном режиме на языке Java. | Реализовывать созданные классы на языке Java.  Разрабатывать, внедрять и адаптировать прикладное программное обеспечение на языке Java. | Отчет по лабораторным работам, тестовый контроль. |
| Владеть | Навыками процедурного программирования, базовыми навыками объектно-ориентированного программирования. | Навыками программирования на объектно**-**ориентированном языке программирования Java.  Навыками разработки, внедрения и адаптации прикладного программного обеспечение на языке Java. | Отчет по лабораторным работам, тестовый контроль, экзамен. |

***Примечание****: Графы 2, 3, …, и т.д. берутся из «Матрицы соответствия компетенций» ООП. Графа 4 – перечисляются оценочные средства, которые затем раскрываются в таблице 6.2.*

## 6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Оценочные средства** | **Уровни сформированности компетенции** | | | |
| Компетентность не сформирована | Базовый уровень компетентности | Достаточный уровень компетентности | Высокий уровень компетентности |
| Опрос | Ответ не структурирован без учета специфики проблемы | Ответ слабо структурирован, не связан с ранее изученным, не выделены существенные признаки проблемы. | Ответ структурирован, оформлен согласно требованиям, однако есть несущественные недостатки. | Ответ структурирован, оформлен согласно требованиям |
| Лабораторная работа | Не выполнена или выполнена с грубыми нарушениями, выводы не соответствуют цели работы. | Выполнена частично или с нарушениями, выводы не соответствуют цели. | Работа выполнена полностью, отмечаются несущественные недостатки в оформлении. | Работа выполнена полностью, оформлена по требованиям. |
| Домашнее задание | Не выполнено или выполнено с грубыми нарушениями, выводы не соответствуют цели работы. | Выполнено частично или с нарушениями, выводы не соответствуют цели. | Работа выполнена полностью, отмечаются несущественные недостатки в оформлении. | Работа выполнена полностью, оформлена по требованиям. |
| Тестовый контроль | Не раскрыт полностью ни один теоретический вопрос, практическое задание не выполнено или выполнено с грубыми ошибками | Теоретические вопросы раскрыты с замечаниями, однако логика соблюдена. Практическое задание выполнено, но с замечаниями | Задания выполнены с несущественными замечаниями | Все задания выполнены правильно |
| Экзамен | Не раскрыт полностью ни один теоретический вопрос, практическое задание не выполнено или выполнено с грубыми ошибками | Теоретические вопросы раскрыты с замечаниями, однако логика соблюдена. Практическое задание выполнено, но с замечаниями: намечен ход выполнения, однако не полно раскрыты возможности выполнения | Работа выполнена с несущественными замечаниями | Работа выполнена полностью, оформлена по требованиям. |

***Примечание.*** *Оценочные средства и критерии формирования компетенции преподаватель определяет самостоятельно в рамках специфики дисциплины. Показатели, шкалы оценивания детализируются в п. 6.4.*

## 6.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

***Образец тестового задания***

**Вопрос 1** (*1 балл*). Рассмотрим пример кода:

try {

// код, бросающий исключения

} catch (RuntimeException e) {

// ...

} catch (Exception e) {

// ...

} catch (Error e) {

// ...

} finally {

// ...

}

Предположим, что во время исполнения программы в теле блока try было выброшено исключение типа Java.lang.IllegalStateException. Какие блоки catch-finally будут выполнены?

*Выберите один или несколько ответов:*

* finally;
* catch (Error e);
* catch (Exception e);
* catch (RuntimeException e).

**Вопрос 2** (*1 балл*). Реализуйте метод sqrt(), вычисляющий квадратный корень числа. В отличие от Math.sqrt(), это метод при передаче отрицательного параметра должен бросать исключение Java.lang.IllegalArgumentException с сообщением “Expected non-negative number, got ?”, где вместо вопросика будет подставлено фактически переданное в метод число.

**Вопрос 3** (*1 балл*). Какого типа исключения нужно объявлять явным образом с помощью оператора throws, который включается в объявление метода, например:

public void f() throws ?

где вместо вопроса следует указать тип исключения(ий).

*Выберите один ответ:*

* Все типы исключений;
* Все типы исключений, кроме RuntimeException;
* Все типы исключений, кроме Error;
* Все типы исключений, кроме RuntimeException и Error;
* Нет необходимости указывать тип исключения.

**Вопрос 4** (*1 балл*). Отметьте неправильные утверждения.

*Выберите один или несколько ответов:*

* Непроверяемые (unchecked) исключения являются наследниками класса Java.lang.Exception;
* Непроверяемые (unchecked) исключения невозможно обработать;
* Проверяемые (checked) исключения обязательно обрабатываются;
* Проверяемые (checked) исключения являются наследниками класса Java.lang.Exception;
* Непроверяемые (unchecked) исключения являются наследниками класса Java.lang.Error.

**Вопрос 5** (*1 балл*). Рассмотрим пример кода:

try {

// ...

} catch (Exception e) {

// ...

} finally {

// ...

}

При каких условиях блок finally не будет выполнен?

*Выберите один или несколько ответов:*

* Если в блоке try выполняется оператор return.
* Если в блоке try выполняется оператор break.
* Если в блоке catch при обработке исключения случилось другое исключение.
* Если во время исполнения блока try виртуальная машина «выполнила недопустимую операцию и будет закрыта».
* Если в блоке try брошено исключение, которое обработано блоком catch.
* Если в блоке try вызван System.exit(0).

***Список экзаменационных вопросов***

1. Цикл разработки программных средств с использованием Java.
2. Минимальный комплект для разработки программ на Java.
3. Средства разработки и системные требования.
4. Основные технические возможности IDE (IDEA, Eclipse, NetBeans).
5. Переменные, константы, область видимости в Java.
6. Примитивные типы данных. Ссылочные типы данных.
7. Логические операторы и арифметические операторы.
8. Преобразования типов. Оболочечные классы.
9. Сложные типы данных. Массивы. Строки.
10. Управляющие конструкции. Оператор if, switch.
11. Операторы цикла. Операторы перехода.
12. Основные принципы ООП. Инкапсуляция, наследование, полиморфизм.
13. Достоинства и недостатки ООП.
14. Понятие объекта и работа с ним.
15. Класс. Описание полей класса. Оператор new.
16. Ссылка this. Оператор super.
17. Описание методов класса. Модификаторы доступа. Конструкторы.
18. Перегрузка и переопределение методов класса.
19. Абстрактные классы. Интерфейсы.
20. Ввод вывод в Java.
21. Элементы графического интерфейса.
22. Контейнеры и элементы управления библиотеки Swing.
23. Java коллекции. Генерики (Java Generics).
24. Сериализация и десериализация объектов в Java. Transient переменные.
25. Исключения и обработка ошибок. Блоки try и catch. Блок finally.
26. Интерфейс Iterator.
27. Потоки ввода-вывода.
28. Работа с сетью в Java.
29. Сокеты. RMI.
30. Функциональные интерфейсы. Лямбда-выражения.

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

***Методические указания по выполнению***

Согласно требованиям ВАК России, провести исследования по темам самостоятельной работы и оформить использованную литературу в соответствии с требованиями ГОСТ Р 7.0.5-2008.

***Пособия и инструменты***

Требования ВАК России по оформлению статей и результатов научного исследований и ГОСТ Р 7.0.5-2008.

В исключительных случаях, когда студент в течение семестра набрал не менее 48 баллов, баллы экзаменационной составляющей могут быть выставлены по усмотрению преподавателя без сдачи экзамена. Основанием для этого являются:

* сделанный на конференции доклад по тематике дисциплины;
* опубликованная или представленная к печати статья по тематике дисциплины, патенты, РИД;
* эффективное участие (занятие призового места) в предметной олимпиаде;
* победа на конкурсе студенческих работ по тематике дисциплины.

## 6.5. Итоговая рейтинговая оценка текущей и промежуточной аттестации студента по дисциплине

В ГБОУВО РК КИПУ используется рейтинговая 100-бальная система оценивания (50 баллов текущего контроля и 50 баллов промежуточного контроля, согласно Положению ГБОУВО РК КИПУ «О балльно-рейтинговой системе оценки знаний и обеспечения качества учебного процесса»). В зачетно-экзаменационную ведомость вносится оценка по четырехбалльной системе. Студент, выполнивший все учебные поручения и набравший в семестре не менее 30 баллов, допускается к зачету или экзамену. Оценка на зачете или экзамене – 30-50 баллов, которые суммируются с баллами семестра. В итоге студент, получивший не менее 60 баллов, считается аттестованным.

По учебным дисциплинам, где итог оценивания уровня знаний студентов предусматривает зачет, максимальная суммарная оценка текущего контроля (модульных контролей) должна составлять 100 баллов. Зачет выставляется во время последнего семинарского (практического, лабораторного) занятия при условии, что суммарная оценка текущей аттестации студента превышает 60 баллов («удовлетворительно» – и выше). Если студент набрал менее 60 баллов, он сдает зачет на последнем практическом занятии.

Итоговая рейтинговая оценка *R* академической успешности студента по дисциплине определяется по формуле:

,

где *Тi* – рейтинговая оценка студента по всем формам текущего контроля;

*Э* – рейтинговая оценка студента по результатам экзамена.

*Использовать для перевода следующую шкалу:*

***Шкала оценивания текущей и промежуточной аттестации студента***

|  |  |  |  |
| --- | --- | --- | --- |
| Уровни формирования компетенции | Сумма баллов по всем формам контроля | Оценка по четырехбалльной шкале | |
| для экзамена, курсового проекта (работы), практики | для зачета |
| Высокий | 90-100 | отлично | зачтено |
| Достаточный | 74-89 | хорошо |
| Базовый | 60-73 | удовлетворительно |
| Компетенция не сформирована | 0-59 | неудовлетворительно | не зачтено |

*Текущий контроль* включает в себя проверку усвоения студентом теоретических знаний и практических умений в ходе изучения учебного материала (устный опрос, тесты и др. виды контроля в соответствии с п. 6.2. в ходе аудиторных занятий).

***Рейтинговая оценка текущего контроля за семестр для студентов ОФО***

|  |  |  |  |
| --- | --- | --- | --- |
| Форма контроля | Уровни формирования компетенций | | |
| Базовый | Достаточный | Высокий |
| 1 | 2 | 3 | 4 |
| Опрос | 2-4 | 4-6 | 6-7 |
| Защита лабораторных работ | 12-13 | 13-14 | 14-15 |
| Тестовый контроль | 12-14 | 14-16 | 16-18 |
| Домашние задания | 4-6 | 6-9 | 9-10 |
| **Общая сумма баллов** | **30-37** | **37-45** | **45-50** |

***Примечание****: в графе 1 формы контроля соответствуют пункту 6.2. Общую сумму баллов по уровням преподаватель может ранжировать в пределах между 30-50 баллами.*

***Рейтинговая оценка промежуточного контроля за семестр***

|  |  |  |  |
| --- | --- | --- | --- |
| Форма контроля | Уровни формирования компетенций | | |
| Базовый | Достаточный | Высокий |
| 1 | 2 | 3 | 4 |
| Теоретические вопросы | 13-15 | 15-18 | 18-20 |
| Практическое задание | 13-15 | 15-18 | 18-20 |
| Тестовые задания | 4-6 | 7-8 | 9-10 |
| **Общая сумма баллов** | **30-36** | **37-44** | **45-50** |

# 7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

*Основная рекомендуемая литература:*

* Блинов И.Н., Романчик В.С. Java. Методы программирования: уч.-мет. пособие. – Минск: Издательство «Четыре четверти», 2013. – 896 с.
* Шилдт Г. Java 8: руководство для начинающих, 6-е изд.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2015. – 720 с.
* Эккель Б. Философия Java. Библиотека программиста, 4-е изд. – СПб.: Питер, 2009. – 640 с.

*Дополнительная рекомендуемая литература, имеющаяся в наличии в фонде Научной библиотеки ГБОУВО РК «КИПУ»:*

* Арнольд К., Гослинг Д. Разработка серверных приложений. – СПб.: Питер, 1997. – 304 с.
* Horstmann, Cay S. Core Java. Volume I. – Fundamentals. Tenth edition. – New York: Prentice Hall, 2016. – 1010 p.

# 8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

*Основные ресурсы:*

* Вязовик Н.А. Программирование на Java [Электронный ресурс] / Н.А. Вязовик. – 2003. – Режим доступа: [http://www.intuit.ru/studies/  
  courses/16/16/info](http://www.intuit.ru/studies/courses/16/16/info).
* Монахов В.В Разработка серверных приложений и среда NetBeans [Электронный ресурс] / В.В. Монахов. – 2009. – Режим доступа: [http://www.intuit.ru/studies/  
  courses/569/425/info](http://www.intuit.ru/studies/courses/569/425/info).
* Свистунов А.Н. Построение распределенных систем на Java [Электронный ресурс] / А.Н. Свистунов. – 2010. – Режим доступа: [http://www.intuit.ru/  
  studies/courses/633/489/info](http://www.intuit.ru/studies/courses/633/489/info).
* Чибриков В. Углубленное программирование на Java [Электронный ресурс] / В. Чибриков. – 2013. – Режим доступа: [http://www.intuit.ru/studies/courses/  
  3711/953/info](http://www.intuit.ru/studies/courses/3711/953/info).

*Дополнительные ресурсы:*

* Открытые информационные научные ресурсы ведущих научных центров и научных журналов.
* Международный электронный архив научных статей: <http://arxiv.org>.
* Научная электронная библиотека eLIBRARY.RU (<http://elibrary.ru>). Крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты более 12 млн. научных статей и публикаций. На платформе elibrary.ru доступны электронные версии более 1400 российских научно-технических журналов, в том числе более 500 журналов в открытом доступе.

*Лицензионные курсы, онлайн-ресурсы, интерактивные занятия:*

|  |  |  |
| --- | --- | --- |
| 1. | Национальный открытый Университет «ИНТУИТ» | <http://intuit.ru> |
| 2. | Free Online Course Materials | MIT OpenCourseWare | <http://ocw.mit.edu> |
| 3. | Free Online Courses From Top Universities | <https://www.coursera.org> |
| 4. | MOOCs/free-online courses | <http://www.udacity.com> |
| 5. | Free online courses from the world's best universities | <http://www.edx.org> |

# 9. Методические указания для обучающихся по освоению дисциплины

## 9.1. Общие рекомендации по самостоятельной работе студентов

Подготовка современного бакалавра предполагает, что в стенах университета он овладеет методологией самообразования, самовоспитания, самосовершенствования. Это определяет важность активизации его самостоятельной работы. С целью организации данного вида учебных занятий необходимо в первую очередь использовать материал лекций и семинаров. Лекционный материал создает проблемный фон с обозначением ориентиров, наполнение которых содержанием производится студентами на семинарских занятиях после работы с учебными пособиями, монографиями и периодическими изданиями.

Самостоятельная работа формирует творческую активность студентов, представление о своих научных и социальных возможностях, способность вычленять главное, совершенствует приемы обобщенного мышления. Самостоятельная работа студентов по дисциплине предполагает более глубокую проработку ими отдельных тем курса, определенных программой. Основными видами и формами самостоятельной работы студентов поданной дисциплине являются: подготовка сообщений и докладов к практическим/семинарским занятиям; выполнение практических заданий; самоподготовка по вопросам; подготовка к дидактическому тесту, экзамену.

Важной частью самостоятельной работы является чтение учебной литературы. Основная функция учебников – ориентировать студента в системе тех знаний, умений и навыков, которые должны быть усвоены по данной дисциплине будущими специалистами. Учебник также служит путеводителем по многочисленным произведениям, ориентируя в именах авторов, специализирующихся на определенных научных направлениях, в названиях их основных трудов. Вторая функция учебника в том, что он очерчивает некий круг обязательных знаний по предмету, не претендуя на глубокое их раскрытие.

Чтение рекомендованной литературы – это та главная часть системы самостоятельной учебы студента, которая обеспечивает подлинное усвоение науки. Читать эту литературу нужно по принципу: «идея, теория, метод в одной, в другой и т. д. книгах». Во всех случаях рекомендуется рассмотрение теоретических вопросов не менее чем по трем источникам. Изучение проблемы по разным источникам – залог глубокого усвоения науки. Именно этот блок, наряду с выполнением практических заданий является ведущим в структуре самостоятельной работы студентов. Вниманию бакалавров предлагаются список литературы, вопросы к семинарам и экзамену, а также практические задания. По желанию они по интересующим вопросам могут написать рефераты, предварительно согласовав тему с преподавателем. Для подготовки к семинарским занятиям преподавателем предлагается ряд вопросов для написания докладов. Требования к оформлению докладов и рефератов такие же, как к оформлению контрольных работ для бакалавров заочного отделения.

Для успешного овладения курсом необходимо выполнять следующие требования:

1) выполнять все домашние задания;

2) посещать занятия, т. к. весь тематический материал взаимосвязан между собой и, зачастую, самостоятельного теоретического овладения пропущенным материалом недостаточно для качественного его усвоения;

3) все рассматриваемые на занятиях вопросы обязательно фиксировать в отдельную тетрадь и сохранять ее до окончания обучения в вузе;

4) проявлять активность при подготовке и на занятиях, т. к. конечный результат овладения содержанием дисциплины необходим, в первую очередь, самому бакалавру;

5) в случаях пропуска занятий по каким-либо причинам обязательно отрабатывать пропущенное преподавателю во время индивидуальных консультаций.

Внеурочная деятельность бакалавра по данной дисциплине предполагает:

– самостоятельный поиск ответов и необходимой информации по предложенным вопросам;

– выполнение заданий;

– выработку умений научной организации труда.

Успешная организация времени по усвоению данной дисциплины во многом зависит от наличия у бакалавра умения самоорганизовать себя и свое время для выполнения предложенных домашних заданий. Объем заданий рассчитан максимально на 2-3 часа в неделю. При этом алгоритм подготовки будет следующим:

1 этап – поиск в литературе теоретической информации по предложенным преподавателем вопросам;

2 этап – осмысление полученной информации, освоение терминов и понятий;

3 этап – составление плана ответа на каждый вопрос;

4 этап – поиск примеров по данной проблематике.

## 9.2. Методические рекомендации по освоению лекционного материала, подготовке к лекциям

Программой дисциплины предусмотрено чтение лекций в различных формах их проведения: проблемные лекции с элементами эвристической беседы, информационные лекции, лекции с опорным конспектированием, лекции-визуализации. На лекциях преподаватель рассматривает вопросы программы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удается осветить в полном объеме, поэтому преподаватель, по своему усмотрению, некоторые вопросы выносит на самостоятельную работу студентов, рекомендуя ту или иную литературу. Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. Во время самостоятельной проработки лекционного материала особое внимание следует уделять возникшим вопросам, непонятным терминам, спорным точкам зрения. Все такие моменты следует выделить или выписать отдельно для дальнейшего обсуждения на семинарском занятии. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в рабочей программе дисциплины.

## 9.3. Методические рекомендации по подготовке к практическим занятиям

Основной целью данного предмета является расширение научного кругозора и формирование практических навыков необходимых **академическому** бакалавру. Отсюда следует, что при подготовке студентов к практическим занятиям по дисциплине нужно не только знакомить студентов с новейшими теориями и методами в прикладной информатике, но и стремиться отрабатывать на практических занятиях полученные умения. Подготовка студентов должна быть ориентирована на глубокое освоение методологии прикладной информатики; Подготовка студентов должна быть ориентирована на глубокое освоение методологии прикладной информатики, разработки программного обеспечения; формирование навыков проектирования, реализации программного обеспечения; формирование умения анализировать возникшую проблему, ставить на её основе исследовательские задачи и подбирать адекватный инструментарий для их решения; формирование стремления к постоянному самосовершенствованию, расширению палитры своего методического инструментария. Практическое занятие – это активная форма учебного процесса в вузе, направленная на умение студентов переработать учебный текст, обобщить материал, развить критичность мышления, отработать практические навыки. В рамках ООП применяются следующие виды практических занятий: семинар-конференция (студенты выступают с докладами, которые тут же и обсуждаются), семинар-дискуссия (научная дискуссия, основанная на поиске материала), обсуждение отдельных вопросов на основе обобщения материала, развернутая беседа в виде плана (при освоении трудного материала), практическая отработка конкретных методов исследования, обсуждение результатов проведенного сравнения, оформление текстового материала в виде таблиц и схем.

Практические занятия предназначены для усвоения материала через систему основных понятий компьютерной науки. Они включают обсуждение отдельных вопросов, разбор трудных понятий и их сравнение в разных научных школах, решение различных задач сферы информационных технологий. Успешная организация времени по усвоению данной дисциплины во многом зависит от наличия у студента умения самоорганизовать себя и свое время для выполнения предложенных домашних заданий. Объем заданий рассчитан максимально на 1-2 часа в неделю.

Подготовка рефератов, сообщений и докладов к семинарским занятиям Доклад, реферат является формой работы, при которой студент самостоятельно готовит сообщение на заданную тему и далее на семинарском занятии выступает с этим сообщением. Целью докладов является более глубокое знакомство с одной из проблем информационных технологий. Доклад должен быть построен таким образом, чтобы наиболее ярко охарактеризовать выбранную проблему и сформировать интерес к ее дальнейшему изучению. Обязательным требование является научное, толерантное и корректное изложение материала. Доклад является элементом промежуточной аттестации и оценивается. В течение семестра каждый студент должен сделать как минимум один доклад, реферат. Если студент за время теоретического обучения не делает доклад, ему необходимо принести письменный текст доклада, реферата на экзамен. В таком случае, в ходе экзамена ему могут быть заданы вопросы по теме доклада. При подготовке к докладам необходимо:

– подготовить сообщение, включающее сравнение точек зрения различных авторов;

– сообщение должно содержать анализ точек зрения, изложение собственного мнения или опыта по данному вопросу, примеры;

– вопросы к аудитории, позволяющие оценить степень усвоения материала;

– выделение основных мыслей, так чтобы остальные студенты могли конспектировать сообщение в процессе изложения.

По усмотрению преподавателя рефераты могут быть представлены на семинарах, а также может быть использовано индивидуальное собеседование преподавателя с бакалавром по выбранной теме.

При разработке реферата используется не менее 3 различных источников.

Реферат должен соответствовать заявленной теме.

Учитывается:

– глубина проработки материала,

– правильность и полнота использования источников.

– оформление реферата.**Требования к выполнению контрольной работы**

Контрольная работа является обязательной частью ФОС по дисциплине. Целью выполнения контрольной работы является более глубокое изучение отдельных вопросов и закономерностей компьютерной науки.

В начале семестра студент изучает требования и рекомендации по выполнению контрольной работы, а также рекомендуемую и дополнительную литературу по дисциплине. В течение семестра преподаватель уточняет и объясняет наиболее сложные вопросы как дисциплины в целом, так и касающиеся отдельных заданий данной контрольной работы. Кроме того, студент может получить дополнительную консультацию преподавателя на кафедре в заранее согласованное время.

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

Для изучения дисциплины «Разработка серверных приложений» используются следующая операционная система и программное обеспечение:

1. Операционная система Windows (XP/Vista/7/8) или Unix (FreeBSD, Linux и др.).
2. Java Development Kit (сокращенно JDK) – бесплатно распространяемый компанией Oracle Corporation (ранее Sun Microsystems) комплект разработчика приложений на языке Java, включающий в себя компилятор Java (Javac), стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java (JRE).
3. Интегрированная среда разработки модульных кроссплатформенных приложений – Netbeans / Eclipse / IntelliJ IDEA.
4. Широкополосный доступ к сети Интернет.

# 11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для проведения лекционных занятий по учебной дисциплине «Архитектура встроенных систем» на достаточно высоком профессиональном уровне целесообразно использовать мультимедийную аудиторию, вместимостью более 80 человек. Аудитория должна состоять из интегрированных инженерных систем с единой системой управления, оснащенная современными средствами воспроизведения и визуализации любой видео и аудио информации, получения и передачи электронных документов. Типовая комплектация мультимедийной аудитории состоит из:

1. Мультимедийный проектор.
2. Автоматизированный проекционный экран.
3. Акустическая система.
4. Интерактивная трибуна преподавателя, включающая тач-скрин монитор с диагональю не менее 22 дюймов.
5. Персональный компьютер (с техническими характеристиками не ниже Intel Core i5-6600k, DDR3 4096Mb, 500Gb).
6. Конференц-микрофон.
7. Беспроводной микрофон.
8. Блок управления оборудованием.
9. Интерфейсы подключения: USB, audio, HDMI.

Интерактивная трибуна преподавателя является ключевым элементом управления, объединяющим все устройства в единую систему, и служит полноценным рабочим местом преподавателя. Преподаватель имеет возможность легко управлять всей системой, не отходя от трибуны, что позволяет проводить лекции, практические занятия, презентации, вебинары, конференции и другие виды аудиторной нагрузки обучающихся в удобной и доступной для них форме с применением современных интерактивных средств обучения, в том числе с использованием в процессе обучения всех корпоративных ресурсов. Мультимедийная аудитория также оснащена широкополосным доступом в сеть интернет. Компьютерное оборудованием имеет соответствующее лицензионное программное обеспечение, в частности необходимо наличие установленных на компьютерах операционных систем Windows (NT/XP/Vista/7/8) или Unix (FreeBSD, Linux и др.).

Компьютерный класс с подключением к системе телекоммуникаций (электронная почта, Интернет), учебные помещения, оснащенные видеотехникой и мультимедийной аппаратурой.

# Оформление комплекта заданий для контрольной работы

**Комплект заданий для контрольной работы**

по дисциплине «Архитектура встроенных систем»

Компетенции: ПК-13.

**Правила выполнения контрольной работы**

Студент должен выполнить 5 заданий контрольной работы (2 теоретических и 3 практических). Работы, выполненные не полностью, а частично, будут возвращены на доработку.

***а) выбор варианта***

Вариант выбирается в соответствии с номером в журнале посещаемости группы. Если количество студентов в группе больше числа вариантов, то номер варианта определяется как остаток от деления номера студента в списке группы на количество вариантов; если остаток 0, то последний вариант.

***б) содержание отчета***

Отчет обязательно должен включать в содержании следующие разделы:

* Титульный лист, оформленный в соответствии со стандартом.
* Содержание.
* Задание на выполнение.
* Необходимый теоретический материал.
* Окончательный вариант исходного кода на языке Java.
* Исходный код класса(ов) с использование которого(ых) производилось тестирование конструкторов и методов реализованных классов.

Оформленный письменно отчет и все выполненные задания должны быть записаны на оптический диск. Работа сдается в печатном и электронном вариантах, вложенных в файл; диск вкладывается в бумажный конверт с указанием фамилии и инициалов студента.

***в) общие требования к оформлению контрольной работы***

1. Работа выполняется печатным способом с использованием компьютера и принтера на одной стороне листа формата А4 через один (1,0) интервал. Ориентация листа: книжная. Тип шрифта (гарнитура) – Times New Roman, кегль (размер) шрифта – 12, выравнивание – по ширине. В работе не допускается использование шрифта разных гарнитур. Для оформления программного кода использовать: тип шрифта (гарнитура) – Courier New, кегль (размер) шрифта – 10, выравнивание – по левой стороне, без абзацев.

2. Абзацы в тексте следует начинать с отступа, равного 12,5 мм.

3. Названия разделов и подразделов выделяются жирным шрифтом. Название раздела располагается посередине строки, а подраздела – от абзаца.

4. Список использованных источников и литературы печатается через 1,0 интервал, каждое название начинается с абзаца.

5. Страницы имеют следующие поля: левое 20 мм, правое – 10 мм, верхнее и нижнее – 15 мм.

6. Все структурные элементы работы начинаются с нового листа.

7. Заголовки располагаются посередине страницы и указываются прописными буквами без кавычек и точки в конце, выделяются полужирным шрифтом. Переносить слова в заголовке не допускается. Заголовки оглавления должны точно повторять заголовки в тексте.

8. Иллюстрации, используемые в тексте работы, размещаются после первой ссылки на них и сопровождаются словами «Рисунок», «Таблица», «Схема», «График» и т. п. Все иллюстрации нумеруются сквозной нумерацией арабскими цифрами в порядке упоминания в тексте (для каждого вида иллюстраций своя нумерация).

9. Приложения должны иметь тематические заголовки и нумеруются арабскими цифрами. Перечень приложений указывается в оглавлении.

10. Все страницы работы, включая приложения, нумеруются по порядку. Первой страницей считается титульный лист, на нем номер не ставится. Нумеруют арабскими цифрами, используя шрифт Times New Roman, кегль 12, с соблюдением сквозной нумерации по всему тексту. Номер проставляется внизу страницы (выравнивание по центру) без точки в конце.

11. Ссылки нумеруются в сквозном порядке арабскими цифрами в пределах каждой части работы (введения, разделов и заключения).

12. Список источников и литературы оформляется в соответствии с ГОСТ 7.1-2003, ГОСТ 7.80-2000, ГОСТ 7.82-2001.

**Задания контрольной работы**

**ЗАДАНИЕ I**

***Исследовать теоретические аспекты вопроса. Ответ должен четко и подробно отражать суть вопроса.***

**Вариант Вопрос**

1. Приведите несколько причин использования языка/платформы Java.
2. Какие главные отличия между Java платформой и другими программными платформами?
3. Какие главные отличия между языком C++ и Java?
4. Что такое Java загрузчики классов и их назначение? Объясните динамическую загрузку классов.
5. Что Вы подразумеваете под полиморфизмом, наследованием, инкапсуляцией и динамическим связыванием?
6. В чем разница между абстрактным классом и интерфейсом, и в каких случаях Вы должны использовать каждый из них?
7. В чем разница перегрузки метода от его переопределения? Можно ли в Java перегружать операции? Если можно, то как, если нет, то почему?
8. Какие главные отличия между ArrayList и LinkedList? Какие главные отличия между HashMap и HashTable?
9. Какое главное отличие между «передачей по значению» и «передачей по ссылке»?
10. Что такое сериализация? Как исключить поле класса из сериализации или что такое transient переменная?
11. Каким образом можно повысить производительность Java I/O операций?
12. Объясните понятие «поверхностное копирование» и «глубокое копирование» объектов в Java? В чем состоит различие между ними?
13. В чем разница между переменной экземпляра и статической переменной? Приведите пример, когда Вы могли бы использовать статическую переменную?
14. Какие модификаторы доступа присутствуют в Java? Каково их предназначение применительно к переменным, методам, классам?
15. Где и как можно использовать закрытый (private) конструктор?

**ЗАДАНИЕ II**

***Исследовать теоретические аспекты вопроса. Ответ должен четко и подробно отражать суть вопроса.***

**Вариант Вопрос**

1. Объектно-ориентированное программирование: инкапсуляция, полиморфизм, наследование.
2. Элементарные типы данных в Java. Логический тип данных. Литералы.
3. Арифметические операции. Операции сравнения и логические операции.
4. Порядок создания объектов. Переменные ссылочного типа и присваивание.
5. Конструкторы. Параметризированные конструкторы.
6. Оператор new. Сборка мусора и методы завершения.
7. Поразрядные операции.
8. Модификаторы доступа в Java.
9. Переопределение методов. Перегрузка методов. Перегрузка конструкторов.
10. Интерфейсы. Реализация интерфейсов.
11. Наследование интерфейсов. Методы интерфейсов, используемые по умолчанию.
12. Использование ключевых слов try и catch. Необработанные исключения.
13. Потоковая организация системы ввода-вывода Java. Байтовые и символьные потоки.
14. Файловый ввод-вывод с использованием символьных потоков.
15. Функциональные интерфейсы. Применение лямбда-выражений.

**ЗАДАНИЕ III**

***Выполнить практическое задание. Привести исходный код класса(ов) и примеры тестирования разработанного приложения.***

Создать классы, спецификации которых приведены ниже. Определить конструкторы и методы **set***Tun***()**, **get***Tun***()**, **toString()**. Определить дополнительно методы в классе, создающем массив объектов. Задать критерий выбора данных и вывести эти данные на консоль. В каждом классе, обладающем информацией, должно быть объявлено несколько конструкторов.

1. **Notebook:** id, Производитель, Модель, Размер (диагональ), Процессор, Объем оперативной памяти, Объем жесткого диска.

Создать массив объектов. Вывести:

*а*) список моделей ноутбуков заданного производителя;

*б*) список ноутбуков заданного размера;

*в*) список ноутбуков с определенным процессором, объемом оперативной памяти и жесткого диска;

*г*) характеристики ноутбука определенной модели.

1. **Student:** id, Фамилия, Имя, Отчество, Дата рождения, Адрес, Телефон, Факультет, Курс, Группа.

Создать массив объектов. Вывести:

*а*) список студентов заданного факультета;

*б*) списки студентов для каждого факультета и курса;

*в*) список студентов, родившихся после заданного года;

*г*) список учебной группы.

1. **Customer**: id, Фамилия, Имя, Отчество, Адрес, Номер кредитной карточки, Номер банковского счета.

Создать массив объектов. Вывести:

*а*) список покупателей в алфавитном порядке;

*б*) список покупателей, у которых номер кредитной карточки находится в заданном интервале.

1. **Patient:** id, Фамилия, Имя, Отчество, Адрес, Телефон, Номер медицинской карты, Диагноз.

Создать массив объектов. Вывести:

*а*) список пациентов, имеющих данный диагноз;

*б*) список пациентов, номер медицинской карты которых находится в заданном интервале.

1. **Abiturient**: id, Фамилия, Имя, Отчество, Адрес, Телефон, Оценки.

Создать массив объектов. Вывести:

*а*) список абитуриентов, имеющих неудовлетворительные оценки;

*б*) список абитуриентов, у которых сумма баллов выше заданной;

*в*) выбрать заданное число *n* абитуриентов, имеющих самую высокую сумму баллов (вывести также полный список абитуриентов, имеющих полупроходную сумму).

1. **Book:** id, Название, Автор(ы), Издательство, Год издания, Количество страниц, Цена, Тип переплета.

Создать массив объектов. Вывести:

*а*) список книг заданного автора;

*б*) список книг, выпущенных заданным издательством;

*в*) список книг, выпущенных после заданного года.

1. **House:** id, Номер квартиры, Площадь, Этаж, Количество комнат, Улица, Тип здания, Срок эксплуатации.

Создать массив объектов. Вывести:

*а*) список квартир, имеющих заданное число комнат;

*б*) список квартир, имеющих заданное число комнат и расположенных на этаже, который находится в заданном промежутке;

*в*) список квартир, имеющих площадь, превосходящую заданную.

1. **TV:** id, Производитель, Модель, Размер (диагональ), Тип телевизора (жидкокристаллический/плазменный), Вес.

Создать массив объектов. Вывести:

*а*) список моделей телевизоров указанного производителя;

*б*) список телевизоров заданного размера;

*в*) список телевизоров определенного типа и диагонали;

*г*) характеристики телевизора определенной модели.

1. **Phone:** id, Фамилия, Имя, Отчество, Адрес, Номер кредитной карточки, Дебет, Кредит, Время городских и междугородных разговоров.

Создать массив объектов. Вывести:

*а*) сведения об абонентах, у которых время внутригородских разговоров превышает заданное;

*б*) сведения об абонентах, которые пользовались междугородной связью;

*в*) сведения об абонентах в алфавитном порядке.

1. **Car:** id, Марка, Модель, Год выпуска, Цвет, Цена, Регистрационный номер.

Создать массив объектов. Вывести:

*а*) список автомобилей заданной марки;

*б*) список автомобилей заданной модели, которые эксплуатируются больше *n* лет;

*в*) список автомобилей заданного года выпуска, цена которых больше указанной.

1. **Product:** id, Наименование, UPC, Производитель, Цена, Срок хранения, Количество.

Создать массив объектов. Вывести:

*а*) список товаров для заданного наименования;

*б*) список товаров для заданного наименования, цена которых не превосходит заданную;

*в*) список товаров, срок хранения которых больше заданного.

1. **Train:** Пункт назначения, Номер поезда, Время отправления, Число мест (общих, купе, плацкарт, люкс).

Создать массив объектов. Вывести:

*а*) список поездов, следующих до заданного пункта назначения;

*б*) список поездов, следующих до заданного пункта назначения и отправляющихся после заданного часа;

*в*) список поездов, отправляющихся до заданного пункта назначения и имеющих общие места.

1. **Bus:** Фамилия и инициалы водителя, Номер автобуса, Номер маршрута, Марка, Год начала эксплуатации, Пробег.

Создать массив объектов. Вывести:

*а*) список автобусов для заданного номера маршрута;

*б*) список автобусов, которые эксплуатируются больше заданного срока;

*в*) список автобусов, пробег у которых больше заданного расстояния.

1. **Airline:** Пункт назначения, Номер рейса, Тип самолета, Время вылета, Дни недели.

Создать массив объектов. Вывести:

*а*) список рейсов для заданного пункта назначения;

*б*) список рейсов для заданного дня недели;

*в*) список рейсов для заданного дня недели, время вылета для которых больше заданного.

1. **Dog:** Кличка, Порода, Возраст, Окрас, Пол.

Создать массив объектов. Вывести:

*а*) список собак заданной породы;

*б*) список собак определенного пола;

*в*) список собак с возрастом не более *n* лет;

*г*) Список собак определенной окраски и породы.

**ЗАДАНИЕ IV**

***Выполнить практическое задание. Привести исходный код класса(ов) и примеры тестирования разработанного приложения.***

Создать консольное приложение, удовлетворяющее следующим требованиям:

* Использовать возможности ООП: классы, наследование, полиморфизм, инкапсуляция.
* Каждый класс должен иметь отражающее смысл название и информативный состав.
* Наследование должно применяться только тогда, когда это имеет смысл.
* При кодировании должны быть использованы соглашения об оформлении кода Java Code Convention ([http://www.oracle.com/technetwork/Java/  
  codeconvtoc-136057.html](http://www.oracle.com/technetwork/java/codeconvtoc-136057.html)).
* Классы должны быть грамотно разложены по пакетам.
* Консольное меню должно быть минимальным.
* Для хранения параметров инициализации можно использовать файлы.

1. **Цветочница.** Определить иерархию цветов. Создать несколько объектов-цветов. Собрать букет (используя аксессуары) с определением его стоимости. Провести сортировку цветов в букете на основе уровня свежести. Найти цветок в букете, соответствующий заданному диапазону длин стеблей.

2. **Новогодний подарок.** Определить иерархию конфет и прочих сладостей. Создать несколько объектов-конфет. Собрать детский подарок с определением его веса. Провести сортировку конфет в подарке на основе одного из параметров. Найти конфету в подарке, соответствующую заданному диапазону содержания сахара.

3. **Электроприбор.** Определить иерархию электроприборов. Включить некоторые в розетку. Подсчитать потребляемую мощность. Провести сортировку приборов в квартире на основе мощности. Найти прибор в квартире, соответствующий заданному диапазону параметров.

4. **Шеф-повар.** Определить иерархию овощей. Сделать салат. Подсчитать калорийность. Провести сортировку овощей для салата на основе одного из параметров. Найти овощи в салате, соответствующие заданному диапазону калорийности.

5. **Мотоциклист.** Определить иерархию амуниции. Экипировать мотоциклиста. Подсчитать стоимость. Провести сортировку амуниции на основе веса. Найти элементы амуниции, соответствующие заданному диапазону параметров цены.

6. **Транспорт.** Определить иерархию подвижного состава железнодорожного транспорта. Создать пассажирский поезд. Подсчитать общую численность пассажиров и багажа. Провести сортировку вагонов поезда на основе уровня комфортности. Найти в поезде вагоны, соответствующие заданному диапазону параметров числа пассажиров.

7. **Авиакомпания.** Определить иерархию самолетов. Создать авиакомпанию. Посчитать общую вместимость и грузоподъемность. Провести сортировку самолетов компании по дальности полета. Найти самолет в компании, соответствующий заданному диапазону параметров потребления горючего.

8. **Таксопарк.** Определить иерархию легковых автомобилей. Создать таксопарк. Подсчитать стоимость автопарка. Провести сортировку автомобилей парка по расходу топлива. Найти автомобиль в компании, соответствующий заданному диапазону параметров скорости.

9. **Мобильная связь.** Определить иерархию тарифов мобильной компании. Создать список тарифов компании. Подсчитать общую численность клиентов. Провести сортировку тарифов на основе размера абонентской платы. Найти тариф в компании, соответствующий заданному диапазону параметров.

10. **Фургон кофе.** Загрузить фургон определенного объема грузом на определенную сумму из различных сортов кофе, находящихся, к тому же, в разных физических состояниях (зерно, молотый, растворимый в банках и пакетиках). Учитывать объем кофе вместе с упаковкой. Провести сортировку товаров на основе соотношения цены и веса. Найти в фургоне товар, соответствующий заданному диапазону параметров качества.

11. **Игровая комната.** Подготовить игровую комнату для детей разных возрастных групп. Игрушек должно быть фиксированное количество в пределах выделенной суммы денег. Должны встречаться игрушки родственных групп: маленькие, средние и большие машины, куклы, мячи, кубики. Провести сортировку игрушек в комнате по одному из параметров. Найти игрушки в комнате, соответствующие заданному диапазону параметров.

12. **Налоги.** Определить множество и сумму налоговых выплат физического лица за год с учетом доходов с основного и дополнительного мест работы, авторских вознаграждений, продажи имущества, получения в подарок денежных сумм и имущества, переводов из-за границы, льгот на детей и материальной помощи. Провести сортировку налогов по сумме.

13. **Счета.** Клиент может иметь несколько счетов в банке. Учитывать возможность блокировки/разблокировки счета. Реализовать поиск и сортировку счетов. Вычисление общей суммы по счетам. Вычисление суммы по всем счетам, имеющим положительный и отрицательный балансы отдельно.

14. **Туристические путевки.** Сформировать набор предложений клиенту по выбору туристической путевки различного типа (отдых, экскурсии, лечение, шопинг, круиз и т. д.) для оптимального выбора. Учитывать возможность выбора транспорта, питания и числа дней. Реализовать выбор и сортировку путевок.

15. **Кредиты.** Сформировать набор предложений клиенту по целевым кредитам различных банков для оптимального выбора. Учитывать возможность досрочного погашения кредита и/или увеличения кредитной линии. Реализовать выбор и поиск кредита.

**ЗАДАНИЕ V**

***Выполнить практическое задание. Привести исходный код класса(ов) и примеры тестирования разработанного приложения.***

Создать программу обработки текста учебника по программированию с использованием классов: *Символ*, *Слово*, *Предложение*, *Абзац*, *Лексема*, *Листинг*, *Знак препинания* и др. Во всех задачах с формированием текста заменять табуляции и последовательности пробелов одним пробелом.

Предварительно текст следует разобрать на составные части, выполнить одно из перечисленных ниже заданий и вывести полученный результат.

1. Найти наибольшее количество предложений текста, в которых есть одинаковые слова.
2. Вывести все предложения заданного текста в порядке возрастания количества слов в каждом из них.
3. Найти такое слово в первом предложении, которого нет ни в одном из остальных предложений.
4. Во всех вопросительных предложениях текста найти и напечатать без повторений слова заданной длины.
5. В каждом предложении текста поменять местами первое слово с последним, не изменяя длины предложения.
6. Напечатать слова текста в алфавитном порядке по первой букве. Слова, начинающиеся с новой буквы, печатать с красной строки.
7. Рассортировать слова текста по возрастанию доли гласных букв (отношение количества гласных к общему количеству букв в слове).
8. Слова текста, начинающиеся с гласных букв, рассортировать в алфавитном порядке по первой согласной букве слова.
9. Все слова текста рассортировать по возрастанию количества заданной буквы в слове. Слова с одинаковым количеством расположить в алфавитном порядке.
10. Существует текст и список слов. Для каждого слова из заданного списка найти, сколько раз оно встречается в каждом предложении, и рассортировать слова по убыванию общего количества вхождений.
11. В каждом предложении текста исключить подстроку максимальной длины, начинающуюся и заканчивающуюся заданными символами.
12. Из текста удалить все слова заданной длины, начинающиеся на согласную букву.
13. Отсортировать слова в тексте по убыванию количества вхождений заданного символа, а в случае равенства – по алфавиту.
14. В заданном тексте найти подстроку максимальной длины, являющуюся палиндромом, т. е. читающуюся слева направо, и справа налево одинаково.
15. В некотором предложении текста слова заданной длины заменить указанной подстрокой, длина которой может не совпадать с длиной слова.

Составитель \_\_\_\_\_\_\_\_\_\_\_\_\_ к.т.н., доцент Абдурайимов Л.Н.

(подпись)

1. Учитываются часы, отведенные на КСР и контроль [↑](#footnote-ref-1)