

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ

Методические указания

по организации самостоятельной работы обучающихся
по дисциплине «Технологии и методы программирования»
для студентов направления подготовки /специальности
10.03.01 Информационная безопасность
шифр и наименование направления подготовки/ специальности

(ЭЛЕКТРОННЫЙ ДОКУМЕНТ)

СОДЕРЖАНИЕ

1. Общие положения	3
2. Требования к уровню освоения содержания дисциплины	3
6. Компетенции обучающегося, формируемые в результате изучения дисциплины.....	3
5. Методические рекомендации для студентов по изучению дисциплины	4
5.1 Использование материала учебно-методического комплекса дисциплины	4
5.2 Работа с литературой	4
6. Методические рекомендации к СРС	7
6.1. Вопросы к экзамену	8
7. Учебно-методическое и информационное обеспечение дисциплины	9

1. Общие положения

Самостоятельная работа студентов (СРС) является существенной частью учебно-воспитательного процесса в высшей школе. Она предназначена не только для овладения конкретной дисциплиной, но и для формирования самостоятельности как одного из важнейших качеств современного специалиста, навыков самостоятельной работы в учебной, научной и профессиональной деятельности, способности работать в нестандартных ситуациях, принимать на себя ответственность, самостоятельно находить конструктивные решения проблем. В современных условиях специалист должен обладать фундаментальными знаниями, профессиональными умениями и навыками, опытом творческой, исследовательской и социально-оценочной деятельности. Решению этой задачи в значительной степени способствует самостоятельная работа студентов.

Самостоятельную работу рассматривают как метод обучения и определяют как средство организации и выполнения студентами определенной деятельности в соответствии с поставленной целью.

Самостоятельная работа студентов находится в тесной взаимосвязи и логически дополняет аудиторную групповую и индивидуальную работу студента с преподавателем.

Основной целью представленных методических указаний по планированию и организации самостоятельной работы является оказание помощи студенту при подготовке к экзамену в процессе изучения учебного материала, закрепления и систематизации знаний.

2. Требования к уровню освоения содержания дисциплины

Целью изучения дисциплины является рассмотрение основных методов и средств разработки, отладки и тестирования ПО, методов коллективной работы по созданию программ, стандартов оформления программной документации.

Задачи дисциплины "Технологии и методы программирования" в соответствии с указанной целью при изучении ставятся следующие:

- ознакомить студентов с современными языками высокого уровня;
- ознакомить с типовыми приемами разработки и записи алгоритмов программ;
- ознакомить с проблемами отладки и тестирования ПО;
- привить навыки работы по созданию программ на языках высокого уровня.

Дисциплина относится к базовой части Б.1.Б.10. Ее освоение происходит в 4 семестре.

3. Место дисциплины в структуре основной образовательной программы

Дисциплина относится к профессиональному циклу, Базовая (общепрофессиональная) часть Б.3.Б.10. Ее освоение происходит в 4 семестре.

4.Связь с предшествующими дисциплинами

Учебная дисциплина «Технологии и методы программирования» базируется на изучении таких дисциплин как: «Языки программирования», «Информатика (спецглавы)», «Информационные технологии».

5.Связь с последующими дисциплинами

Изучение дисциплины «Технологии и методы программирования» позволит подготовиться к усвоению последующих дисциплин таких как: «Объектно-ориентированное программирование», «Методы проектирования локальных сетей».

6. Компетенции обучающегося, формируемые в результате изучения дисциплины

Индекс	Формулировка:
--------	---------------

ПК-2	способностью применять программные средства системного, прикладного и специального назначения
------	---

6. Методические рекомендации для студентов по изучению дисциплины

6.1 Использование материала учебно-методического комплекса дисциплины

На первом этапе необходимо ознакомиться с рабочей программой дисциплины, в которой рассмотрено содержание тем дисциплины лекционного курса, взаимосвязь тем лекций с лабораторными занятиями, темы и виды самостоятельной работы. По каждому виду самостоятельной работы предусмотрены определённые формы отчетности.

Технологическая карта самостоятельной работы студента

Код реализуемой компетенции	Вид деятельности студентов	Итоговый продукт самостоятельной работы	Средства и технологии оценки*	Объем часов
ПК-2	Подготовка к лабораторным работам	Индивидуальное задание	Отчет устный	22
ПК-2	Самостоятельное изучение литературы по темам 1-14.	Конспект	Собеседование	12
ПК-2	Подготовка к практическим занятиям	Индивидуальное задание	Отчет устный	14
Итого за 5 семестр				38
Итого				38
Код реализуемой компетенции	Вид деятельности студентов	Итоговый продукт самостоятельной работы	Средства и технологии оценки*	Объем часов
ПК-2	Подготовка к лабораторным работам	Индивидуальное задание	Отчет устный	22
ПК-2	Самостоятельное изучение литературы по темам 1-14.	Конспект	Собеседование	12
ПК-2	Подготовка к практическим занятиям	Индивидуальное задание	Отчет устный	14
Итого за 4 семестр				38
Итого				38

6.2 Работа с литературой

Для успешного освоения дисциплины, необходимо самостоятельно детально изучить представленные темы по рекомендуемым источникам информации:

№ п/п	Темы для самостоятельного изучения	Рекомендуемые источники информации (№ источника)			
		Основная	Дополнительная	Методическая	Интернет-ресурсы
Раздел 1. Теоретические основы технологии программирования					

1	<p>ТЕМА 1. Основные элементы среды разработки. Основные понятия объектно-ориентированного программирования. Среда программирования. Главное меню. Панели инструментов. Инспектор объектов. Дерево объектов. Форма. Размещение объектов на форме. Настройка свойств объектов. Объектная модель языка программирования. Классы. Объекты. Поля. Методы. Свойства. Компоненты. Инкапсуляция. Наследование. Абстракция. Полиморфизм.</p>	1	1-2	1-3	1-2
	<p>ТЕМА 2. Надежное программное средство как продукт технологии программирования. Исторический и социальный контекст программирования. Источники ошибок в программных средствах. Программа как формализованное описание процесса обработки данных. Программное средство. Неконструктивность понятия правильной программы. Надежность программного средства. Технология программирования как технология разработки надежных программных средств. Технология программирования и информатизация общества. Интеллектуальные возможности человека. Неправильный перевод как причина ошибок в программных средствах. Модель перевода. Основные пути борьбы с ошибками.</p>	1	1-2	1-3	1-2
	<p>ТЕМА 3. Общие принципы разработки программных средств. Специфика разработки</p>	1	1-2	1-3	1-2

	<p>программных средств. Жизненный цикл программного средства. Понятие качества программного средства. Обеспечение надежности – основной мотив разработки программных средств. Методы борьбы со сложностью. Обеспечение точности перевода. Преодоление барьера между пользователем и разработчиком. Контроль принимаемых решений. остановка задачи и спецификация программы. Дружественность, жизненный цикл программы</p>				
	<p>ТЕМА 4. Внешнее описание программного средства. Назначение внешнего описания программного средства и его роль в обеспечении качества программного средства. Определение требований к программному средству. Спецификация качества программного средства. Функциональная спецификация программного средства. Методы контроля внешнего описания программного средства.</p>	1	1-2	1-3	1-2
	<p>ТЕМА 5. Архитектура программного средства Понятие архитектуры программного средства. Основные классы архитектур программных средств. Архитектурные функции. Контроль архитектуры программных средств.</p>	1	1-2	1-3	1-2
Раздел 2. Технологии программирования					
2	<p>ТЕМА 6. Разработка структуры программы и модульное программирование.</p>	1	1-2	1-3	1-2
	<p>ТЕМА 7. Разработка программного модуля.</p>	1	1-2	1-3	1-2

ТЕМА 8. Тестирование и отладка программного средства.	1	1-2	1-3	1-2
ТЕМА 9. Обеспечение функциональности и надежности программного средства.	1	1-2	1-3	1-2
ТЕМА 10. Обеспечение качества программного средства.	1	1-2	1-3	1-2
ТЕМА 11. Документирование программных средств.	1	1-2	1-3	1-2
ТЕМА 12. Управление разработкой и аттестация программного средства.	1	1-2	1-3	1-2
ТЕМА 13. Объектный подход к разработке программных средств.	1	1-2	1-3	1-2
ТЕМА 14. Компьютерная поддержка разработки и сопровождения программных средств.	1	1-2	1-3	1-2

7. Методические рекомендации к СРС

Общепризнанными формами СРС, предусматривающие письменное их изложение и оценочный контроль, являются курсовые работы, рефераты и эссе, которые по содержанию помогают глубже усвоить теоретические положения изучаемой дисциплины.

Реферат (лат. referre – докладывать, сообщать) относится к форме письменного доклада на определенную тему, включающий обзор соответствующих литературных и других источников.

Реферат является одной из форм самостоятельной работы студентов и одновременно формой контроля качества усвоения материала курса, умения самостоятельно находить и излагать проблемные вопросы по изучаемой дисциплине.

Подготовка реферата является важной составной частью учебного процесса, приобщает студентов к исследовательской работе, обогащает опытом, умением и знаниями, позволяет реализовать требования государственного образовательного стандарта.

Тематика реферата предлагается преподавателем или согласовывается с ним на индивидуальной консультации. Студент может выбрать любую тему из приведенного перечня или предложить новую, не предусмотренную тематикой.

В процессе написания реферата студент должен обстоятельно изучить рекомендованные литературные источники, ознакомиться с постановкой аналитической работы на конкретном предприятии и на этой основе показать, что он детально усвоил содержание основных вопросов выбранной темы, умеет самостоятельно применять полученные знания к комплексному решению определенной аналитической задачи.

Требования, которым должен отвечать реферат:

- реферат должен быть выполнен на высоком профессиональном уровне, при его подготовке должны быть использованы последние законодательные акты, инструктивные материалы, литературные источники, материалы лекционного курса и практические навыки;
- при подготовке реферата практический материал и литературные источники должны быть увязаны с избранной темой, актуализированы на текущее время, позволять полностью ее раскрыть;

- реферат должен быть правильно оформлен, написан разборчиво, грамотно и аккуратно.

Титульный лист выполняется по традиционной форме, принятой в университете. На нем должны быть указаны полное наименование университета, кафедра, дисциплина по которой он подготовлен, фамилия имя отчество студента с указанием специальности, группы, формы обучения, полное название темы в соответствии с заданием, должность, ученое звание преподавателя и его фамилия и инициалы, год написания.

После него следует план реферата с указанием номеров страниц основного текста, а далее - непосредственное изложение каждого вопроса плана. Нельзя ограничиваться простым пересказом прочитанного материала, необходимо изложить собственное понимание существа вопроса. Цитаты следует приводить только в тех случаях, когда они служат аргументом при изложении какого-либо тезиса или являются объектом критики.

Обязательными элементами реферата кроме основного содержательного текста, должны быть введение и заключение. В конце работы приводится перечень использованной литературы, электронных и нормативных источников, при этом сначала следует указать законодательные и нормативные акты, инструктивные материалы, а затем - специальную литературу в алфавитном порядке. Завершать реферат могут приложения, дополняющие и информационно подкрепляющие его основной, содержательный текст.

Процесс подготовки и защиты реферата включает следующие этапы:

- выбор темы;
- составление плана;
- изучение вопросов, намеченных в плане;
- подбор литературы и ознакомление с ее содержанием;
- подбор практического материала по теме реферата;
- оформление реферата;
- проверка реферата преподавателем и его оценка.

При получении отзыва на реферат студент обязан устранить содержащиеся в ней замечания и дополнить работу, в случае необходимости, теоретическими или практическими материалами. Каждый студент устно защищает реферат тому преподавателю, который его проверял. Во время защиты студент кратко излагает важнейшие положения работы и выводы, к которым он пришел. Затем он отвечает на вопросы, поставленные в рецензии или задаваемые преподавателем устно. По результатам защиты в рецензии проставляется окончательная оценка реферата.

Реферат должен быть выполнен на компьютере. Общий объем реферата (не считая приложений) должен составлять не менее 30 листов формата А4 (210x297 мм), шрифт - Times New Roman, 14, межстрочный интервал одинарный. Поля; верхнее — 2 см, нижнее - 2 см, левое - 2,5 см, правое - 1,5 см.

На последней странице работы ставятся подпись студента и дата.

Эссе (*фр. essai*), относится «к жанру публицистики (критики), представляющий общее или предварительное соображение по какому-либо поводу или о каком-либо предмете».

Как и реферат, является одной из форм самостоятельной работы студента, относится к разновидности письменных работ, выполненных студентом по раскрытию конкретного теоретического вопроса изучаемого курса, или отражающее мнение студента на тот или иной дискуссионный вопрос, поясняющий практическую ситуацию, тот или иной пример из реальной практики. Требования к объему не ограничены, но должны быть достаточными для раскрытия существа рассматриваемого вопроса.

Оформляется эссе, как и реферат, на компьютере. По оформлению это менее заформализованный документ, имеющий титульный лист, аналогичный автореферату, изложение вопроса по существу и список источников, положенных в основу его написания.

7.1. Вопросы к экзамену

1. Подпрограммы. Процедуры. Функции.
2. Рекурсия. Рекурсивные подпрограммы.
3. Файлы. Тестовые файлы.
4. Сложные типы данных. Записи.
5. Модули. Модульное программирование.
6. Основные принципы объектно-ориентированного программирования.
7. Основные понятия объектно-ориентированного программирования. Классы. Объекты.
8. Создание таблиц в Delphi. Построение диаграмм.
9. Применение компонентов – списков в приложениях Delphi.
10. Методы, компоненты Delphi для работы с файлами.
11. Консольные приложения, их создание в Delphi.
12. Статические и динамические переменные. Объявление и инициализация указателей.
13. Обращение к переменной с помощью указателя. Проблемы, связанные с указателями. Стандартные процедуры и функции для работы с динамической памятью.
14. Информационные структуры.
15. Простые списки (на основе массивов). Динамические массивы Delphi. Изменение размеров простых списков: добавление и удаление элементов.
16. Связанные списки. Двусвязные списки.
17. Стек. Изменение размера стека: добавление и удаление элементов.
18. Очереди. Изменение размера очереди: добавление и удаление элементов.
19. Деревья. Бинарные деревья. Изменение размера бинарного дерева: добавление и удаление элементов. Поиск элемента.
20. Динамически компонуемая библиотека DLL, ее применение.
21. Потоки данных. Файловые потоки данных TFileStream.
22. Потоки данных. Потоки данных работы с памятью THandleStream.
23. Технология Drag&Drop.
24. Технология Drag&Drock.
25. Программные потоки Threads.
26. Унифицированный язык моделирования UML.
27. Основные характеристики UML.
28. Основные понятия языка UML: пакеты, подсистемы, представления, модели
29. Представления в UML, соответствующие им типы диаграмм и понятий.
30. Классификаторы и отношения, их функции и обозначения.
31. Структурное представление модели: статическое представление модели. Классы, и отношения между классами, диаграммы классов, их построение в среде BDS 2006 и JBuilder 2006.
32. Структурное представление модели: Представление использования, диаграммы вариантов использования.
33. Структурное представление модели: Представление реализации и развертывания, диаграмма компонентов, диаграмма развертывания.
34. Динамическое поведение модели: Представление в виде конечного автомата, диаграмма состояний.
35. Динамическое поведение модели: Представление деятельности, диаграмма деятельности.
36. Динамическое поведение модели: Представление взаимодействия, диаграмма последовательности, диаграмма кооперации.
37. Управление моделью: Представление управления моделью, пакеты и подпакеты, отношения между ними.

7. Учебно-методическое и информационное обеспечение дисциплины

7.1 Основная литература:

1. Болдырев, Д. В. Технология программирования. Структуры данных : учебное пособие : Направление подготовки 230400 - Информационные системы и технологии, 230700 - Прикладная информатика, Бакалавриат / Д. В. Болдырев ; Сев.-Кав. федер. ун-т. - Ставрополь : СКФУ, 2013. - 195 с. : ил. - Прил.: с. 148
2. Котов, О. М. Язык С : краткое описание и введение в технологии программирования / О.М. Котов. - Екатеринбург : Издательство Уральского университета, 2014. - 209 с. - ISBN 978-5-7996-1094-4

7.2. Дополнительная литература:

1. Основы алгоритмизации и программирования : лабораторный практикум : Направление подготовки 230400.62 – Информационные системы и технологии. Профили подготовки «Информационные системы в бизнесе», «Безопасность информационных систем». Бакалавриат / сост. Е. И. Николаев ; Сев.-Кав. федер. ун-т. - Ставрополь : СКФУ, 2015. - 211 с. : ил.
2. Братченко, Н. Ю. Основы программирования графических объектов средствами OpenGL в MS Visual Studio C++ : учебно-метод. пособие : Направление подготовки 210700.62 – Инфокоммуникационные технологии и системы связи. Профиль подготовки «Сети связи и системы коммутации» / Н. Ю. Братченко ; Сев.-Кав. федер. ун-т. - Ставрополь : СКФУ, 2014. - 88 с.

7.3 Методическая литература:

1.Флоринский О.С. Методические указания по выполнению лабораторных работ по дисциплине «Технологии и методы программирования».

2.Флоринский О.С. Методические указания по выполнению практических работ по дисциплине «Технологии и методы программирования».

7.4 Интернет-ресурсы:

1. <http://www.intuit.ru> – сайт дистанционного образования в области информационных технологий
2. <http://window.edu.ru> – образовательные ресурсы ведущих вузов

7. 5 Программное обеспечение

1. Microsoft Office;
2. Embarcadero RAD Studio 2010;
3. Microsoft Visual Studio.