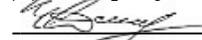


МИНОБРНАУКИ РОССИИ
Ярославский государственный университет им. П.Г. Демидова

Кафедра вычислительных и программных систем

УТВЕРЖДАЮ

Декан факультета ИВТ

 Д.Ю. Чалый

« 23 » мая 2023 г.

Рабочая программа дисциплины
«Практикум ЭВМ по основам программирования»

Направление подготовки
02.03.02 Фундаментальная информатика и информационные технологии

Профиль
«Информатика и компьютерные науки»

Квалификация выпускника
Бакалавр

Форма обучения
очная

Программа рассмотрена
на заседании кафедры
от 21 апреля 2023 г.,
протокол № 8

Программа одобрена НМК
факультета ИВТ
протокол № 6 от
28 апреля 2023 г.

Ярославль

1. Цели освоения дисциплины

Дисциплина «Практикум ЭВМ по основам программирования» обеспечивает приобретение знаний и умений в соответствии с государственным образовательным стандартом, и является одним из основных предметов, позволяющих формировать навыки владения современными информационными технологиями и способствующих развитию алгоритмического мышления у студентов.

Целью изучения дисциплины «Практикум ЭВМ по основам программирования» является освоения принципов структурного программирования и изучение средств языка С и на практических занятиях и лабораторных работах по программированию.

2. Место дисциплины в структуре ОП бакалавриата

Дисциплина «Практикум ЭВМ по основам программирования» относится к вариативной части ОП бакалавриата.

Знания и навыки, полученные при ее изучении, используются учащимися при изучении последующих дисциплин, таких как «Операционные системы», «Языки и методы программирования», «Языки программирования и методы трансляции», «Практикум на ЭВМ по языкам программирования», «Практикум на ЭВМ по объектно-ориентированному программированию»

3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП бакалавриата.

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Формируемая компетенция (код и формулировка)	Индикатор достижения компетенции (код и формулировка)	Перечень планируемых результатов обучения
Профессиональные компетенции		
ПК-2 Способен к разработке требований и проектированию программного обеспечения	ПК-2.1 Знает современные методы разработки и реализации алгоритмов обработки данных	Знать: <ul style="list-style-type: none">– понятия и средства языка С;– структуры данных, алгоритмы и методы их обработки;– принципы разработки программ и отдельных программных модулей; Уметь: <ul style="list-style-type: none">– разрабатывать прикладные программы на языке С;– осуществлять процессы отладки и тестирования. Владеть: <ul style="list-style-type: none">– навыками ООП;– навыками применения интегрированных сред программирования

4. Объем структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 2 зачетные единицы, 72 часа.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах)						Формы текущего контроля успеваемости Форма промежуточной аттестации (по семестрам)
			лекции	практические	лабораторные	консультации	аттестационные испытания	самостоятельная работа	
			Контактная работа						
1	Основные понятия и приемы работы в интегрированных системах разработки	1			2				
2	Функции ввода-вывода.	1			4				
3	Линейные программы. Циклы.	1			4				
4	Одномерные массивы.	1			2			7	задания для самостоятельной работы
5	Двумерные массивы	1			2			7	задания для самостоятельной работы
6	Строки и операции над ними.	1			2			7	задания для самостоятельной работы
7	Функции.	1			4				
8	Односвязные и двусвязные списки.	1			4			8	задания для самостоятельной работы
9	Основы файловой системы.	1			2				
10	Файловый ввод/вывод.	1			2				
11	Динамические структуры данных	1			6			8,7	задания для самостоятельной работы
	Всего за 1 семестр				34		0,3	37,7	зачет
	Всего				34		0,3	37,7	

Содержание разделов дисциплины

1. Основные понятия и приемы работы в интегрированных системах разработки.
 - 1.1. Создание проекта в среде программирования Visual Studio.
 - 1.2. Работа с проектом в среде программирования Visual Studio.
2. Функции ввода-вывода.
 - 2.1. Консольный ввод-вывод.
 - 2.2. Библиотека функций ввода-вывода.
3. Линейные программы. Циклы.
 - 3.1. Условный оператор.
 - 3.2. Операторы цикла.
4. Одномерные массивы.
 - 4.1. Особенности использования одномерных массивов. Структуры данных одномерных массивов. Структуры хранения одномерных массивов.
 - 4.2. Динамические массивы.
5. Двумерные массивы.
 - 5.1. Особенности использования двумерных массивов. Структуры данных двумерных массивов. Структуры хранения двумерных массивов.
 - 5.2. Динамические массивы. Треугольные и разреженные матрицы.
6. Строки и операции над ними.
7. Функции.
 - 7.1. Особенности применения функций.
 - 7.2. Параметры функции и способы их передачи.
 - 7.3. Рекурсивные функции.
8. Односвязные и двусвязные списки.
 - 8.1. Организация структуры связанных списков. Достоинства и недостатки линейных списков. Особенности применения в программных приложениях.
 - 8.2. Операции над линейными списками.
9. Основы файловой системы.
 - 9.1. Общие сведения. Последовательные и библиотечные файлы. Файлы прямого доступа.
 - 9.2. Работа с файлами.
10. Файловый ввод/вывод.
 - 10.1. Ввод-вывод потока. Буферизированный и форматированный ввод-вывод.
 - 10.2. Ввод-вывод нижнего уровня. Небуферизированный и неформатированный ввод-вывод.
11. Динамические структуры данных.
 - 11.1. Общие сведения о бинарных деревьях. Представление деревьев в программных приложениях.
 - 11.2. Разновидности бинарных деревьев. Применение различных деревьев при разработке приложений.

5. Образовательные технологии, используемые при осуществлении образовательного процесса

Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.

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

В процессе осуществления образовательного процесса используются: для разработки документов, презентаций, для работы с электронными таблицами

OfficeStd 2013 RUS OLP NL Acdmc 021-10232

LibreOffice (свободное)

- для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next");
- для проведения лабораторных занятий ОС Windows XP или Linux и интегрированные среды программирования Visual Studio или NetBeans.

7. Перечень основной и дополнительной учебной литературы, ресурсов информационно- телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

а) основная литература:

1. Лагутина, Н. С., Разработка программных приложений : практикум для студентов, обучающихся по направлению Фундаментальная информатика и информационные технологии / Н. С. Лагутина, Ю. А. Ларина, А. М. Васильев; Яросл. гос. ун-т., Ярославль, ЯрГУ, 2014, 71 с.
2. Лагутина, Н. С., Разработка программных приложений [Электронный ресурс] : практикум для студентов, обучающихся по направлению Фундаментальная информатика и информационные технологии / Н. С. Лагутина, Ю. А. Ларина, А. М. Васильев; Яросл. гос. ун-т., Ярославль, ЯрГУ, 2014, 71 с. <http://www.lib.uniyar.ac.ru/edocs/iuni/20140402.pdf>

б) дополнительная литература:

1. В.В. Васильчиков. Библиотечные функции turboC и BorlandC. – Ярославль, ЯрГУ, 2002.
2. Подбельский В.В., Фомин С.С. Программирование на языке Си: учебное пособие для вузов - 2-е изд., доп. - М.: Финансы и статистика, 2003.
3. Лагутина, Н. С., С++. Примеры и задачи : практикум / Н. С. Лагутина ; Яросл. гос. ун-т, Ярославль, ЯрГУ, 2011, 47с

в) ресурсы сети «Интернет»

1. Электронная библиотека учебных материалов ЯрГУ
(http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php).
2. Информационная система "Единое окно доступа к образовательным ресурсам" (<http://www.edu.ru> (раздел Учебно-методическая библиотека)) или по прямой ссылке (<http://window.edu.ru/library>).
3. Информация по языкам программирования, операционным системам, примеры программ: (www.firststeps.ru)

8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине «Практикум на ЭВМ по основам программирования»

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

- учебные аудитории для проведения лабораторных занятий;
- учебные аудитории для проведения групповых и индивидуальных консультаций,
- учебные аудитории для проведения текущего контроля и промежуточной аттестации;
- помещения для самостоятельной работы;
- помещения для хранения и профилактического обслуживания технических средств обучения. Помещения для лабораторных занятий и самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации. Число посадочных мест в лекционной аудитории больше либо равно списочному составу потока, а в аудитории для практических занятий (семинаров) – списочному составу группы обучающихся.

Автор(ы) :

Старший преподаватель кафедры ВПС Ю.А. Ларина

**Приложение №1 к рабочей программе дисциплины
«Практикум ЭВМ по основам программирования»**

**Фонд оценочных средств
для проведения текущей и промежуточной аттестации
студентов по дисциплине**

**1. Типовые контрольные задания или иные материалы,
необходимые для оценки знаний, умений, навыков и (или) опыта
деятельности,**

характеризующих этапы формирования компетенций

**1.1 Контрольные задания и иные
материалы, используемые в процессе текущей
аттестации**

Задания для самостоятельной работы

1. Задачи к разделам 1.2-1.4 из практикума «Разработка программных приложений» (Лагутина Н.С., Ларина Ю.А., Васильев А.М.: практикум. Ярославль, ЯрГУ, 2014).
2. Лабораторные работы 1-6 из учебного пособия «Основы программирования на языке С» (Васильчиков В.В., Лагутина Н.С., Ларина Ю.А.: - Ярославль.: ЯрГУ, 2006.)

Критерии оценивания

Оценка	Критерии
Отлично	Правильно построен алгоритм решения задачи. Программа реализована и выполняется в соответствии с поставленной задачей и правильно выполняется для всех тестов. Разработаны и выполняются полные наборы тестов для проверки работы программы. Студент легко ориентируется в коде работы, вносит поправки, применяет и объясняет изучаемые элементы языка программирования. Используются эффективные методы решения
Хорошо	Правильно построен алгоритм решения задачи. Программа реализована и выполняется в соответствии с поставленной задачей. Программа правильно выполняется для большинства тестов, но необходимы доработки программного кода. Разработаны и выполняются наборы тестов для проверки работы программы. Студент ориентируется в коде работы, вносит поправки, возможно не для всех заданных вопросов. Студент применяет и объясняет основные изучаемые элементы языка программирования (технологии программирования), возможно неполное понимание всех нюансов применяемых средств программирования. Используются эффективные методы решения.
Удовлетворительно	Алгоритм решения задачи построен правильно за исключением отдельных небольших частей решения. Архитектурное решение

	<p>задачи в основном правильное. Программа реализована и выполняется, но не для всех тестов, доработка программного кода не всегда приводит к желаемому результату.</p> <p>Разработаны и выполняются не полные наборы тестов для проверки работы программы. Студент ориентируется в коде работы, но затрудняется внести поправки в случае достаточно сложных вопросов. Студент применяет, но не объясняет основные изучаемые элементы языка программирования (технологии программирования). Используются неэффективные методы решения</p>
Неудовлетворительно	<p>Задача решена неверно или решена совсем другая задача.</p> <p>Программа не выполняется или выполняется неверно для большинства тестов. Отсутствует понимание кода программы и применяемых средств программирования. Код является дубликатом уже существующего решения. Студент не может написать код для решения простых задач.</p>

1.2 Список вопросов и (или) заданий для проведения промежуточной аттестации

Список заданий к зачету

- Для последовательности целых чисел посчитать общую сумму и количество элементов данной последовательности, которые расположены между первым четным и последним нечетным элементом последовательности.
- Произвести преобразование последовательности символов, заменив каждую букву в последовательности на следующую букву латинского алфавита.
- В одномерном массиве символов определить количество цифр, расположенных между двумя звездочками.
- Дана прямоугольная матрица вещественных чисел. Построить по ней другую матрицу путем удаления строки с максимальным номером, в которой все элементы положительны и упорядочены по возрастанию.
- Дана прямоугольная матрица вещественных чисел. Построить по ней другую матрицу путем удаления столбца с минимальным номером, в котором все элементы отрицательны и упорядочены по убыванию.
- Дана квадратная матрица целых чисел. Построить по ней другую матрицу, элементы которой получаются симметричным отражением элементов исходной матрицы относительно главной диагонали.
- Дана квадратная матрица целых чисел. Осуществить циклический сдвиг строк матрицы на заданное количество позиций вверх или вниз, в зависимости от выбранного режима.
- Задана последовательность идентификаторов. Найти и напечатать все идентификаторы с четным числом цифр в них.
- Задано предложение, состоящее из слов, разделенных произвольным количеством пробелов. Написать функцию, которая находит и печатает все самые длинные слова в предложении.
- Задано предложение, состоящее из слов, разделенных произвольным количеством пробелов. Написать функцию, которая распечатывает слова в предложении так, чтобы они заканчивались в заданной позиции экрана.
- Дан односвязный список. Найти и удалить все элементы списка, информационная часть которых равна их порядковому номеру.
- Дан двусвязный список. Переставьте все четные положительные элементы в начало списка.

13. Дан односвязный список. Преобразовать его таким образом, чтобы получился еще один список, в который переставлены все отрицательные элементы из первого списка.
14. Написать программу сбора статистики использования служебных слов языка С в файле с исходным кодом.
15. Написать программу, которая переписывает исходный текстовый файл в результирующий переверачивая все слова задом наперед.
16. Написать программу, которая читает текстовый файл и удаляет все симметричные слова.
17. Написать программу сжатия и последующего восстановления файла.
18. Описать структуру с именем Note содержащую следующие поля: Ф.И.О., номер телефона. Написать программу, которая читает соответствующие данные из файла, строит в памяти список и в процессе его обработки выводит на экран запрашиваемую информацию.
19. В разреженной матрице найти и обнулить все минимальные элементы.
20. Используя динамические структуры данных, написать программу моделирующую заполнение гибкого диска.

Пример задания на зачет.

1. Произвести преобразование последовательности символов, заменив каждую букву в последовательности на следующую букву латинского алфавита.
2. Дана квадратная матрица целых чисел. Построить по ней другую матрицу, элементы которой получаются симметричным отражением элементов исходной матрицы относительно главной диагонали.
3. Задано предложение, состоящее из слов, разделенных произвольным количеством пробелов. Написать функцию, которая находит и печатает все самые длинные слова в предложении.
4. Дан односвязный список. Преобразовать его таким образом, чтобы получился еще один список, в который переставлены все отрицательные элементы из первого списка.

Показатели и критерии, используемые при выставлении оценки

Номер задачи	Критерии	Шкала оценивания
1-4	<p>Знать понятия и средства языка С, структуры данных, алгоритмы и методы их обработки, принципы разработки программ и отдельных прог-раммных модулей</p> <p>Уметь разрабатывать прикладные программы на языке С, осуществлять процессы отладки и тестирования.</p> <p>Владеть навыками структурного программирования, навыками применения интегрированных сред программирования</p>	<p>0 баллов – студент полностью не верно решил задачу, не ответил на поставленные вопросы;</p> <p>1 балл – студент решил поставленную задачу с ошибкой, которую может исправить, отвечает на вопросы большей частью правильно, но может допускать незначительные ошибки;</p> <p>2 балла – студент полностью правильно решил поставленную задачу, грамотно и аргументировано отвечает на поставленные вопросы.</p>

Максимальное количество баллов по и– 8 баллов

Набранное количество баллов соответствует оценке за зачет :

- менее 5 баллов по — оценка «незачтено»,
- от 5 до 7 баллов по — оценка «зачтено» пороговый уровень сформированности компетенции.

Методические указания по оцениванию зачета.

Итоговая оценка по дисциплине «Практикум ЭВМ по основам программирования» формируется в два этапа.

Первый этап – оценивание работы студента в течение семестра за выполнение самостоятельных лабораторных работ. Если на этом этапе все лабораторные работы выполнены в срок и средний балл за текущую аттестацию больше четырех студенту может быть выставлена итоговая оценка «зачтено» автоматом досрочно.

Второй этап – проведение зачета. Для получения оценки «зачтено», студент должен выполнить все лабораторные работы на положительную оценку, возможно не полностью в срок. При выполнении этого условия считается итоговой оценкой, полученной при проведении зачета.

2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

2.1 Шкала оценивания сформированности компетенций и ее описание

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

Пороговый уровень - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

Продвинутый уровень - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

Высокий уровень - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций»

3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объёме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;
- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;
- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;
- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;

- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

3.2 Описание процедуры выставления оценки

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка «зачтено» или «не зачтено».

Показатели и критерии, используемые при выставлении оценки подробно описаны в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций».

Пороговый уровень формирования компетенций соответствует оценке «зачтено» за выполнение лабораторных работ в течение семестра и сдачу зачета в конце семестра.

Оценка «зачет» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Приложение №2 к рабочей программе дисциплины «Практикум ЭВМ по основам программирования»

Методические указания для студентов по освоению дисциплины

Для успешного освоения дисциплины очень важно решение достаточно большого количества задач, как в аудитории, так и самостоятельно в качестве домашних заданий. Примеры решения задач разбираются и обсуждаются на лабораторных занятиях. Основная цель решения задач

– помочь усвоить фундаментальные понятия языка С и основы концепции структурного программирования.

Лабораторные занятия со студентами первого курса в силу их меньшей самостоятельности предполагают более тесный контакт с преподавателем. По этой причине практические занятия по данной дисциплине проводятся в небольших группах студентов, что предполагает большую индивидуализацию процесса обучения. Предоставляется больше возможностей для обратной связи между преподавателем и студентом. При возникающих затруднениях студент в любой момент может задать вопрос по теме и немедленно получить ответ. По наиболее сложным для изучения темам преподавателем проводятся консультации

С целью организации обратной связи и стимуляции более активной работы студентов на практических занятиях предлагаются к обсуждению различные варианты программной реализации задач, поставленных перед аудиторией. Студенты, общаясь между собой, производят их анализ и выработку оптимального решения ("групповое творческое мышление"). Наряду с этим происходит выявление и разбор наиболее типичных ошибок, возникающих в процессе написания программ у начинающих программистов.

Большое внимание должно быть уделено выполнению домашней работы. В качестве заданий для самостоятельной работы дома студентам предлагаются задачи, аналогичные разобранным на лабораторных занятиях. При этом обучающимся предоставляется возможность выбрать свой темп обучения, так как для индивидуального решения предлагаются задачи различного уровня сложности, что позволяет сделать процесс обучения более интенсивным.

Для проверки и контроля усвоения теоретического материала, приобретенных практических навыков программирования на языке С, в течение обучения проводятся мероприятия текущей аттестации в виде проверки заданий для самостоятельной работы. Процесс сдачи выполненного задания проходит индивидуально для каждого студента.

В конце семестра студенты сдают зачет. Зачет по итогам семестра выставляется по результатам выполнения лабораторных работ, а так же письменного зачета в конце семестра. Для освоения вопросов, изучаемых в процессе освоения дисциплины «Практикум на ЭВМ по основам программирования» студенту необходимо уделить особенное внимание самостоятельной работе по овладению основами программирования. Посещение аудиторных занятий является необходимым, так как здесь студент приобретает необходимые для дальнейшей самостоятельной работы знания и навыки. Без упорных и регулярных занятий в течение семестра сдать зачет по итогам изучения дисциплины студенту будет сложно.

Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине

Для самостоятельной работы особенно рекомендуется использовать учебную литературу, указанную в разделе № 7 данной рабочей программы.

Также для подбора учебной литературы рекомендуется использовать широкий спектр интернет-ресурсов:

1. Электронно-библиотечная система «Университетская библиотека online» (www.biblioclub.ru) - электронная библиотека, обеспечивающая доступ к наиболее востребованным материалам-первоисточникам, учебной, научной и художественной литературе ведущих издательств (*регистрация в электронной библиотеке – только в сети университета. После регистрации работа с системой возможна с любой точки доступа в Internet.).
2. Для самостоятельного подбора литературы в библиотеке ЯрГУ рекомендуется использовать:
 1. Личный кабинет (http://lib.uniyar.ac.ru/opac/bk_login.php) дает возможность получения online доступа к списку выданной в автоматизированном режиме литературы, просмотра и копирования электронных версий изданий сотрудников университета (учеб. и метод. пособия, тексты лекций и т.д.) Для работы в «Личном кабинете» необходимо зайти на сайт Научной библиотеки ЯрГУ с любой точки, имеющей доступ в Internet, в пункт меню «Электронный каталог»; пройти процедуру авторизации, выбрав вкладку «Авторизация», и заполнить представленные поля информации.
 2. Электронная библиотека учебных материалов ЯрГУ (http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php) содержит более 2500 полных текстов учебных и учебно-методических материалов по основным изучаемым дисциплинам, изданных в университете. Доступ в сети университета, либо по логину/паролю.
3. Электронная картотека «Книгообеспеченность» (http://www.lib.uniyar.ac.ru/opac/bk_bookreq_find.php) раскрывает учебный фонд научной библиотеки ЯрГУ, предоставляет оперативную информацию о состоянии книгообеспеченности дисциплин основной и дополнительной литературой, а также цикла дисциплин и специальностей. Электронная картотека «Книгообеспеченность» доступна в сети университета и через Личный кабинет.