Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»
Дата и время: 2024-02-21 00:00:00
471088fad29a3b30e244e722abc56b1ab55c9d5024ddef0e75e03a5b6fdf6436

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кемеровский государственный университет»
Новокузнецкий институт (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»

Факультет физико-математический и технолого-экономический
Профилирующая кафедра теории и методики преподавания информатики

УТВЕРЖДАЮ
Декан ФМ и ТЭФ
_____ И.И. Тимченко
«24» марта 2017г.

Рабочая программа дисциплины (модуля)
Б1.В.ОД.9 Практикум по решению задач на ЭВМ

Направление подготовки
44.03.01 Педагогическое образование

Направленность (профиль) подготовки
Информатика

Программа: ***академический бакалавриат***

Квалификация (степень) выпускника
бакалавр

Форма обучения
заочная

Год набора 2016

Новокузнецк, 2017

Лист внесения изменений

Сведения об утверждении:

утверждена Ученым советом факультета
(протокол Ученого совета факультета № _6_ от _3_.03__.2016_)
на 20_16___ год
Одобрена на заседании методической комиссии
протокол методической комиссии факультета № _6_ от 18._02_.2016_)
Одобрена на заседании обеспечивающей кафедры
протокол № _7_ от 16._03_.2016_) М.С.Можаров *(Ф. И.О. зав. кафедрой)* / _____ *(подпись)*

Изменения по годам:

На 2017_ год

утвержден (а) Ученым советом факультета
(протокол Ученого совета факультета № 7 от 16.03.2017)
на 2017 год набора
Одобрен (а) на заседании методической комиссии
протокол методической комиссии факультета № 7 от 15.03.2017)
Одобрен (а) на заседании обеспечивающей кафедры ТиМПИ
протокол № 8 от 02.03.2017) Можаров М.С. (Ф. И.О. зав. кафедрой) / _____ (подпись)

СОДЕРЖАНИЕ

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

2. Место дисциплины в структуре ООП

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

3.1. Объём дисциплины (модуля) по видам учебных занятий (в часах)

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

4.1. Разделы дисциплины (модуля) и трудоемкость по видам учебных занятий (в академических часах)

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

6.1. Паспорт фонда оценочных средств по дисциплине (модулю)

6.2. Типовые контрольные задания или иные материалы

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

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

а) основная учебная литература:

б) дополнительная учебная литература:

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля)

9. Методические указания для обучающихся по освоению дисциплины (модуля)

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

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

12. Иные сведения и (или) материалы

12.1. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

12.2 Занятия, проводимые в интерактивных формах

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

В результате освоения ООП бакалавриата / специалитета / магистратуры (*выбрать)* обучающийся должен овладеть следующими результатами обучения по дисциплине (модулю):

Коды компетенции	**Результаты освоения ООП** ***Содержание компетенций****	**Перечень планируемых результатов обучения по дисциплине**
СПК-2	способен использовать математический аппарат, методологию программирования и со-временные компьютерные технологии для реализации аналитических и технологических решений в области программного обеспечения и компьютерной обработки информации	Знать: методы и приемы формализации и алгоритмизации задач; синтаксис языков программирования (Алгоритмический язык, Basic, Pascal, Python, C, Java, Prolog, Lisp), особенности программирования на выбранном языке, стандартные библиотеки языка программирования; структуры данных и алгоритмы решения типовых задач, области и способы их применения; методологии разработки программного обеспечения; технологии программирования; методы и приемы отладки программного кода, типы и форматы сообщений об ошибках, предупреждений; Уметь: использовать методы и приемы формализации и алгоритмизации поставленных задач; использовать функциональные возможности компиляторов, трансляторов, отладчиков и интегрированных сред разработки для написания и отладки программного кода; применять стандартные алгоритмы в соответствующих областях; применять выбранные языки программирования для написания программного кода; использовать выбранную среду программирования и средства системы управления базами данных; выявлять ошибки в программном коде, применять методы и приемы отладки программного кода, интерпретировать сообщения об ошибках и предупреждения; Владеть: навыками составления формализованных описаний решений поставленных задач; навыками разработки алгоритмов решения поставленных задач; опытом применения выбранных языков программирования для написания программного кода; владеть методами анализа, проверки и отладки исходного программного кода; интерфейсом и функциональными возможностями Case-средств для структурного и объектно-ориентированного проектирования; современными формализованными математическими, информационно-логическими и логико-семантическими моделями и методами представления, сбора и обработки информации; методами программирования и навыками работы с математическими пакетами для решения практических задач хранения и обработки информации.

2. Место дисциплины в структуре ООП бакалавриата

Данная дисциплина относится к обязательным дисциплинам вариативной части блока Б1 и изучается на 4 курсе в 8 семестре.

Освоение данной дисциплины является необходимой основой для последующего изучения дисциплин вариативной части профессионального цикла, прохождения педагогической практики.

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

Общая трудоемкость (объем) дисциплины (модуля) составляет __3___ зачетных единиц (ЗЕТ), ___108_____ академических часов.

3.1. Объём дисциплины (модуля) по видам учебных занятий (в часах)

Объём дисциплины	**Всего часов**	
	для очной формы обучения	для заочной /очно-заочной формы обучения
Общая трудоемкость дисциплины	108	108
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)		
Аудиторная работа (всего**):	32	12
в т. числе:		
Лекции		
Семинары, практические занятия	32	
Практикумы		
Лабораторные работы		12
Занятия в интерактивной форме	8	6
Внеаудиторная работа (всего**):		
В том числе, индивидуальная работа обучающихся с преподавателем:		
Курсовое проектирование		
Групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем		
Творческая работа (эссе)		
Самостоятельная работа обучающихся (всего**)	76	92
Вид промежуточной аттестации обучающегося (зачет / экзамен****)	Зачет с оценкой	Зачет с оценкой 4

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

4.1. Разделы дисциплины (модуля) и трудоемкость по видам учебных занятий (в академических часах)

для очной формы обучения

№ п/п	Раздел дисциплины	Общая трудоёмкость *(часах)*	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах)			Формы текущего контроля успеваемости
			аудиторные учебные занятия		самостоятельная работа обучающихся	
		всего	лекции	семинары, практические занятия		
1.	Одномерные массивы	12		4	8	Решение индивидуальных задач
2.	Двумерные массивы	12		4	8	Решение индивидуальных задач
3.	Строки	12		4	8	Решение индивидуальных задач
4.	Файлы	12		4	8	Решение индивидуальных задач
5.	Сортировка массивов и строк	12		4	8	Контрольные вопросы, лабораторная работа
6.	Подпрограммы: процедуры и функции	12		4	8	Контрольные вопросы, лабораторная работа
7.	Графика	12		4	8	Решение индивидуальных задач
8.	Рекурсия	12		4	8	Решение индивидуальных задач
	Всего:	**108**		**32**	**76**	

4.2 Содержание дисциплины (модуля), структурированное по темам (разделам)

№ п/п	Наименование раздела дисциплины	Содержание
Темы лабораторных занятий		
1	Одномерные массивы.	Ввод и вывода элементов массива. Простейшие

№ п/п	Наименование раздела дисциплины	Содержание
	вычисления с элементами массива.	
2	Одномерные массивы. Поиск минимального (максимального) элемента в массиве и его индекса. Сортировки элементов массива.	
3	Одномерные массивы. Нахождение суммы элементов массива.	
3	Двумерные массивы. Ввод и вывода элементов массива. Простейшие вычисления с элементами массива.	
4	Двумерные массивы. Поиск минимального (максимального) элемента в массиве и его индекса.	
5	Двумерные массивы. Нахождение суммы элементов массива.	
6	Строки. Поиск символа в строке. Удаления символов в строке.	
7	Строки. Подсчет количества слов в строке.	
8	Строки. Поиск и удаление слов.	
9	Текстовые файлы. Работа с текстовыми файлами, чтение и запись.	
10	Текстовые файлы. Удаление, вставка строк.	
11	Текстовые файлы. Перестановка строк.	
12	Сортировка массивов и строк.	
13	Сортировка массивов и строк.	
14	Подпрограммы: процедуры.	
15	Подпрограммы: функции.	
16	Графика и циклы.	
17	Графика и подпрограммы.	
18	Графика. Построение графиков функций.	
19	Рекурсия. Построение.	
20	Рекурсия. Вычисления.	

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Вопросы для самоконтроля по темам:

«Сортировка массивов и строк».

1) Методы сортировок массива: метод пузырька, Шелла, метод вставки, метод выборки, метод двоичных вставок, метод слияний (фон Неймана).

2) Процедура упорядочивания пирамидальной сортировкой.

3) Приемы сортировки строк.

«Подпрограммы: процедуры и функции»:

1) Передача параметров процедурам и функциям.

2) Документация процедур и функций.

3) Отличия функции от процедуры.

4) Три вида областей определения, характеризующих доступность переменной. Инкапсуляция.

5) Реализация диалога с пользователем.

Содержание контрольных мероприятий: проводится в форме решения практической задачи по темам пройденного курса.

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

6.1. Паспорт фонда оценочных средств по дисциплине (модулю)

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции* (или её части) / и ее формулировка – *по желанию*	наименование оценочного средства
1.	Одномерные массивы	СПК-2	Лабораторная работа
2.	Двумерные массивы	СПК-2	Индивидуальные задачи по теме
3.	Строки	СПК-2	Индивидуальные задачи по теме
4.	Текстовые файлы	СПК-2	Индивидуальные задачи по теме
5.	Сортировка массивов и строк	СПК-2	Контрольные вопросы, лабораторная работа
6.	Подпрограммы: процедуры и функции	СПК-2	Контрольные вопросы, лабораторная работа
7.	Графика	СПК-2	Индивидуальные задачи по теме
8.	Рекурсия	СПК-2	Индивидуальные задачи по теме
9.	Зачет		

6.2. Типовые контрольные задания или иные материалы

6.2.1. Зачет

1) типовые вопросы (задания)

Примеры индивидуальных задач по теме «Текстовые файлы». Создайте приложение для обработки текстового файла в среде Gambas.

1. Дан текстовый файл *f*. Записать строки файла *f* в файл *g*. Порядок слов в строках файла *g* должен быть обратным по отношению к порядку слов в строках исходного файла.

2. Дан текстовый файл *f*. Записать строки файла *f* в файл *g*, удаляя при этом из всех слов, состоящих из нечетного числа символов, символ, находящийся посередине слова.

3. Переписать из текстового файла *f* в файл *g* все слова, являющиеся палиндромами (”перевертышами”), разделяя их пробелами и разбивая на строки, содержащие по 5 слов.

4. Дан текстовый файл *f*. Определить, сколько в нем имеется слов, состоящих из одного, двух, трех и т. д. символов.

5. Дан текстовый файл *f*. Переписать из него в файл *g* все слова, состоящие не менее, чем из трех символов и в которых второй и предпоследний символы совпадают между собой. Слова разделять запятой.

6. Дан текстовый файл *f*. Вывести на экран порядковый номер и содержимое строки этого файла, в которой встречается наибольшее количество идущих подряд пробелов.

7. Дан текстовый файл *f*. Подсчитать в нем количество слов, у которых первый и последний символы совпадают между собой.

8. Дан текстовый файл *f*. Переписать его содержимое построчно в файл *g*, упорядочив по алфавиту слова каждой строки исходного файла.

9. Дан текстовый файл *f*. Записать в перевернутом виде строки файла *f* в файл *g*. Порядок строк в файле *g* должен быть обратным по отношению к порядку строк исходного файла.

10. Дан текстовый файл *f*. Удалить из него все однобуквенные слова и лишние пробелы. Результат записать в файл *g*.

11. Написать программу, которая построчно печатает содержимое непустого текстового файла *t*, переворачивая при этом слова в строках.

12. Даны текстовый файл, строка *s*. Вывести на экран все строки файла *f*, содержащие в качестве фрагмента строку *s*.

2) критерии оценивания компетенций (результатов)

Результат оценивается в зависимости от правильности выполнения практического задания. Весьма важным в данном случае является временной фактор. Студент должен справляться с решением индивидуальной задачи по теме в рамках лабораторного занятия.

3) описание шкалы оценивания

Правильность выполнения практического задания оценивается по трехбальной шкале: полностью правильно (2 балла), выполнено с недочетом (1 балл), выполнено полностью неверно (0 баллов).

6.2.2 Наименование оценочного средства* (*в соответствии с таблицей 6.1***)**

6.2.1. Зачет

10. типовые вопросы (задания)

Примеры индивидуальных задач по теме «Текстовые файлы». Создайте приложение для обработки текстового файла в среде Lazarus.

2. Дан текстовый файл *f*. Записать строки файла *f* в файл *g*. Порядок слов в строках файла *g* должен быть обратным по отношению к порядку слов в строках исходного файла.
3. Дан текстовый файл *f*. Записать строки файла *f* в файл *g*, удаляя при этом из всех слов, состоящих из нечетного числа символов, символ, находящийся посередине слова.
4. Переписать из текстового файла *f* в файл *g* все слова, являющиеся палиндромами (”перевертышами”), разделяя их пробелами и разбивая на строки, содержащие по 5 слов.
5. Дан текстовый файл *f*. Определить, сколько в нем имеется слов, состоящих из одного, двух, трех и т. д. символов.
6. Дан текстовый файл *f*. Переписать из него в файл *g* все слова, состоящие не менее, чем из трех символов и в которых второй и предпоследний символы совпадают между собой. Слова разделять запятой.

7. Дан текстовый файл *f*. Вывести на экран порядковый номер и содержимое строки этого файла, в которой встречается наибольшее количество идущих подряд пробелов.
8. Дан текстовый файл *f*. Подсчитать в нем количество слов, у которых первый и последний символы совпадают между собой.
9. Дан текстовый файл *f*. Переписать его содержимое построчно в файл *g*, упорядочив по алфавиту слова каждой строки исходного файла.
10. Дан текстовый файл *f*. Записать в перевернутом виде строки файла *f* в файл *g*. Порядок строк в файле *g* должен быть обратным по отношению к порядку строк исходного файла.
11. Дан текстовый файл *f*. Удалить из него все однобуквенные слова и лишние пробелы. Результат записать в файл *g*.
12. Написать программу, которая построчно печатает содержимое непустого текстового файла *t*, переворачивая при этом слова в строках.
13. Даны текстовый файл, строка *s*. Вывести на экран все строки файла *f*, содержащие в качестве фрагмента строку *s*.

11. критерии оценивания результатов

Результат оценивается в зависимости от правильности выполнения практического задания. Весьма важным в данном случае является временной фактор. Студент должен справляться с решением индивидуальной задачи по теме в рамках лабораторного занятия.

12. описание шкалы оценивания

Правильность выполнения практического задания оценивается по трехбальной шкале: полностью правильно (2 балла), выполнено с недочетом (1 балл), выполнено полностью неверно (0 баллов).

6.2.2 Наименование оценочного средства* (*в соответствии с таблицей 6.1***)**

4) типовые задания (вопросы) - образец

Структура лабораторной работы представлена следующим образом:

- выполнение лабораторной работы (фронтальное, индивидуальное).

Лабораторная работа. Построение графиков функций, представленных в явном виде, параметрическом виде, в полярных координатах.

В среде Lazarus используются компоненты, обладающие свойством Canvas, а сам компонент рассматривается как прямоугольная сетка, состоящая из отдельных точек. Свойством Canvas обладают такие компоненты как: форма (класс Tform), область рисования (класс TpaintBox), растровая картинка (класс Timage), панель (класс Tpanel) и др.

Рассмотрим пример построения графика функции непосредственно на форме в среде Lazarus, заданного в параметрическом виде.

Задача 1. Построить график функции. Уравнение кривой задано в параметрическом виде:

$$\begin{cases} x = R(t - \sin t), \\ y = R(1 - \cos t). \end{cases}$$

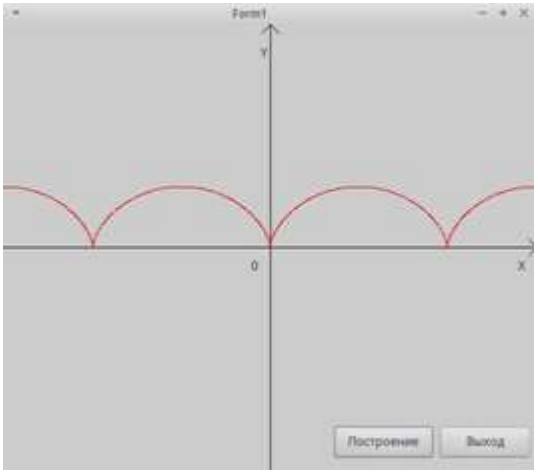


Рис. 1. График функции в среде Lazarus.

На форме разместим две кнопки класса Tbutton из вкладки Standard. Построение графика будет осуществляться по нажатию на кнопку (“Построение”). Программный код которой приведем ниже:

```
procedure TForm1.Button1Click(Sender: TObject);
var r, d, x01, y01 :integer;
 t,x,y:real;
begin
 r:=30; // Задаем значение переменной, отвечающей за масштабирование графика функции.
 d:=10; //  Задаем длину направляющих стрелок декартовой системы координат.
 x01:=Form1.Width div 2; // Определим начало координат.
 y01:=Form1.Height div 2;
 Form1.Canvas.Line(0,y01,x01*2,y01); // Построим ось абсцисс.
 Form1.Canvas.Line(x01,0,x01,y01*2); // Построим ось ординат.
 Form1.Canvas.Line(x01,0,x01+d,d); //Построим направляющие для оси оридинат.
 Form1.Canvas.Line(x01,0,x01-d,d);
 Form1.Canvas.Line(x01*2,y01,x01*2-d,y01+d); //Построим направляющие для оси абсцисс.
 Form1.Canvas.Line(x01*2,y01,x01*2-d,y01-d);
 Form1.Canvas.TextOut(x01-d,d*2,'Y'); // Подпишем оси и начало координат.
 Form1.Canvas.TextOut(x01*2-d*2,y01+d,'X');
 Form1.Canvas.TextOut(x01-d*2,y01+d,'0');
 t:=-10; // Зададим начальное значение параметра.
 While t<=10  do
 begin
 x:=r*(t-sin(t)); // Определим координату точки кривой, заданной в параметрическом представлении.
 y:=-r*(1-cos(t));
 t:=t+0.01; // Увеличим значение параметра с шагом 0,01.
Form1.Canvas.Pixels[trunc(x01+x),trunc(y01+y)]:=clred; //Построим точку красного цвета с вычисленными координатами.
end;
end;
```

В следующей задаче построим график функции, используя отрезки.

Задача 2. Построить график функции. Уравнение кривой задано в явном виде: $y = 1 - 2x\sqrt{\frac{x}{-x}}$. При этом необходимо помнить об ограничениях, накладываемых на интервал значений для x, то есть по корнем не может быть отрицательного значения и знаменатель не может принимать значение 0.

Ниже приведен листинг программного кода для построения графика функции, заданного

а параметрическом виде.

```
procedure TForm1.Button2Click(Sender: TObject);
 var x0,y0,d,m: integer; i,y,y1,y2,x,x1:real;
 begin
d:=10;
x1:=0;
PaintBox1.Canvas.Brush.Color:=clwhite;// задаем цвет фона заливки прямоугольника
PaintBox1.Canvas.FillRect(Rect(0,0,PaintBox1.Width,PaintBox1.Height)); // рисуем закрашенный прямоугольник
x0:=PaintBox1.Width div 2; // находим центр координат
y0:=PaintBox1.Height div 2;
PaintBox1.Canvas.Line(x0,0,x0, y0*2); // рисуем ось ординат
PaintBox1.Canvas.Line(0,y0, x0*2, y0); // ось абсцисс
PaintBox1.Canvas.Line(x0,0,x0-d,d); // направляющие для оси ординат
PaintBox1.Canvas.Line(x0,0,x0+d,d);
PaintBox1.Canvas.Line(x0*2-d,y0-d,x0*2,y0); // направляющие для оси абсцисс
PaintBox1.Canvas.Line(x0*2-d,y0+d,x0*2,y0);
PaintBox1.Canvas.Pen.Color:=clred;
PaintBox1.Canvas.TextOut(x0*2-d*2,y0+d*2,'X'); // выводим текст
PaintBox1.Canvas.TextOut(x0-d*2,d*2,'Y');
PaintBox1.Canvas.TextOut(x0-d*2,y0+d*2,'0');
m:=80;
PaintBox1.Canvas.MoveTo(trunc(x0),trunc(y0));
i:=x1;
While i<=1 do
begin
y1:=y0-trunc((1-2*i)*sqrt(i/abs(1-i))*m);
i:=i+0.001;
x:=x0+i*m;
PaintBox1.Canvas.LineTo(trunc(x),trunc(y1));
end;
PaintBox1.Canvas.MoveTo(trunc(x0),trunc(y0));
i:=x1;
While i<=1 do
begin
y2:=y0+trunc((1-2*i)*sqrt(i/abs(1-i))*m);
i:=i+0.001;
x:=x0+i*m;
PaintBox1.Canvas.LineTo(trunc(x),trunc(y2));
end;
   end;
```

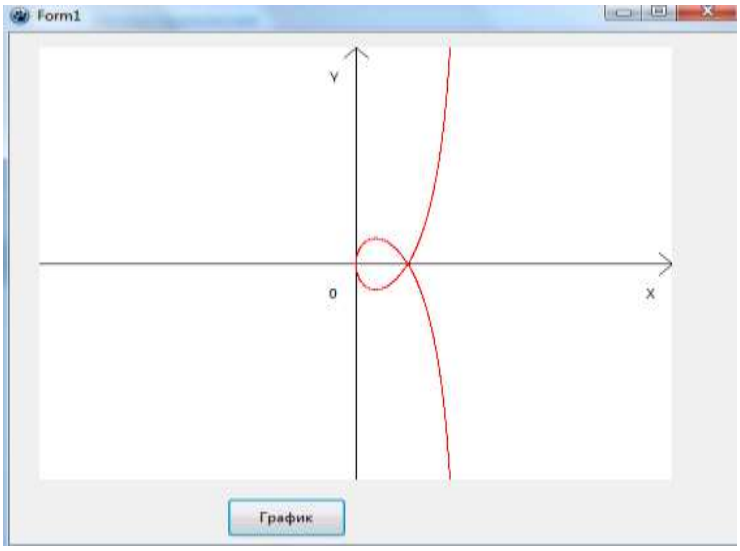


Рис. 2. График функции в среде Lazarus.

Выполните самостоятельно следующие задачи:

Задача 3. Создайте изображение графика функции, заданного в явном виде: $y = \pm (1 - 2x)\sqrt{\frac{x}{1-x}}$.

Задача 4. В графическом режиме нарисовать график функции *Улитка Паскаля.* Уравнение кривой в полярных координатах: $\rho = a\cos j + l$, $a>0$, $l>0$. Рассмотреть случаи, когда $l<a$, $l=a$, $l>a$.

5) критерии оценивания результатов

Результаты *опросов* оцениваются по доле правильных ответов на вопросы и задания. Выполнение практической части лабораторной работы оценивается следующим образом: оценивается объем и правильность выполнения работы.

6) описание шкалы оценивания.

Лабораторная работа выполнена в полном объеме -1 балл, выполнена частично – 0,5 балла, не выполнена – 0 баллов.

6.3 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования ЗУН

В конце изучения всех тем подводятся итоги работы студентов на практических занятиях путем суммирования всех заработанных баллов.

Поскольку студент выполняет различные виды работ, получает за них не только максимальное, но и минимальное количество баллов, то получаемый результат (сумма) целиком зависит от его активности в течение семестра. Выполняющий все задания студент значительно облегчает себе сдачу экзаменационного теста, поскольку набирает большое количество баллов предыдущими видами работ.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

а) основная учебная литература:

1. Введение в структурное программирование [Текст] : учебное пособие / М.

С. Можаров, Г. Н. Бойченко ; Министерство образования и науки РФ ; Кузбасская государственная педагогическая академия. - Новокузнецк : [КузГПА], 2014. - 203 с. - Библиогр.: с. 203 (13 назв.). - Дар автора. - ISBN 978-5-85117-759-0. (ФМФ и ТЭФ - 30 экз).

2. Борисов С.В. Введение в среду визуального программирования Turbo Delphi: МГТУ им. Н.Э. Баумана (Московский государственный технический университет имени Н.Э. Баумана), 2011. - 99 с. Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=52432
3. Осипов В.П. Практикум по программированию на языке Delphi. Часть 1. - МГТУ им. Н.Э. Баумана (Московский государственный технический университет имени Н.Э. Баумана), 2010. - 111с. Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=52406

б) дополнительная учебная литература:

4. Кауфман В.Ш. Языки программирования. Концепции и принципы. - М.: "ДМК Пресс", 2010. - 464 с. Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=1270
5. Медведик В.И. Практика программирования на языке Паскаль (задачи и решения). - М.: "ДМК Пресс", 2013. - 590 с. Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=58700
6. Канцедал С.А. Алгоритмизация и программирование : Учебное пособие / С.А. Канцедал. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 352 с.: ил.; 60x90 1/16. - (Профессиональное образование). (переплет) ISBN 978-5-8199-0355-1, 500 экз. Режим доступа: http://znanium.com/bookread.php?book=429576
7. Голицына О.Л. Языки программирования : Учебное пособие / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. - 2-е изд., перераб. и доп. - М.: Форум, 2010. - 400 с.: ил.; 60x90 1/16. - (Профессиональное образование). (переплет) ISBN 978-5-91134-442-9, 1500 экз. Режим доступа: http://znanium.com/bookread.php?book=226043.

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

Электронно-библиотечные системы: ЭБС Издательства Лань http://e.lanbook.com/, ЭБС «znanium.com» http://znanium.com, ЭБС «Университетская библиотека online» http://biblioclub.ru

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Вид учебных занятий	Организация деятельности студента
Лабораторная работа	Вузовская подготовка специалистов должна обеспечивать приобретение ими не только знаний, но и умений использовать полученные знания на практике. Это требование и положено в основу целей и методов

	проведения лабораторных работ по вышеуказанной учебной дисциплине. Лабораторные работы предлагаются в соответствии с рабочей программой в рамках каждой темы.
Подготовка к зачету	Подготовка к зачету предполагает изучение рекомендуемой литературы и других источников, конспектов лекций, повторение материалов практических занятий.

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

1. Применяется системное и прикладное программное обеспечение при выполнении лабораторных работ.
2. Используются электронные ресурсы и ресурсы Интернет для подготовки к занятиям;
3. Консультирование студентов и контроль выполнения лабораторных работ осуществляется посредством электронной почты.

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для проведения практических занятий необходима аудитория на 20 мест.

Для использования электронных ресурсов в рамках проведения лабораторных занятий необходима персональная компьютерная техника с доступом в Интернет.

При выполнении лабораторных работ применяется операционная система Linux и объектно-ориентированная среда разработки программ Gambas.

12. Иные сведения и (или) материалы

12.1. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Особенности реализации программы курса для инвалидов и людей с ограниченными возможностями здоровья зависит от состоянии их здоровья и конкретных проблем, возникающих в каждом отдельном случае.

- При организации образовательного процесса для слабослышащих студентов от преподавателя курса требуется особая фиксация на собственной артикуляции. Говорить следует немного громче и четче.
- На занятиях преподавателю требуется уделять повышенное внимание специальным профессиональным терминам, а также к использованию профессиональной лексики. Для лучшего усвоения слабослышащими специальной терминологии необходимо каждый раз писать на доске используемые термины и контролировать их усвоение.
- В процессе обучения рекомендуется использовать разнообразный наглядный материал. Все лекции курса снабжены компьютерными мультимедийными презентациями.
- В процессе работы со слабовидящими студентами педагогическому работнику следует учитывать, для усвоения информации слабовидящим требуется большее количество повторений и тренировок по сравнению с лицами с нормальным зрением.

- Информацию необходимо представлять в том виде, в каком ее мог бы получить слабовидящий обучающийся: крупный шрифт (16 - 18 пунктов). Следует предоставить возможность слабовидящим использовать звукозаписывающие устройства и компьютеры во время занятий по курсу. При лекционной форме занятий студенту с плохим зрением следует разрешить пользоваться диктофоном - это его способ конспектировать. Не следует забывать, что все записанное на доске должно быть озвучено.
- В работе с маломобильными обучающимися предусматривается возможность консультаций посредством электронной почты.

12.2 Занятия, проводимые в интерактивных формах

п/п	Раздел, тема дисциплины	Объем аудиторной работы в интерактивных формах по видам занятий (час.)			Формы работы
		Лекц.	Практич	Лабор.	
1	Решение задач по программированию				проблемная лекция
				2	работа в малых группах
2	Решение сложных задач по программированию				проблемная лекция
				2	работа в малых группах,
3	Решение олимпиадных задач по программированию				проблемная лекция
				4	работа в малых группах
	ИТОГО по дисциплине:			8	

Составитель (и): Буяковская И.А., доцент
(*фамилия, инициалы и должность преподавателя (ей)*)

Макет рабочей программы дисциплины (модуля) разработан в соответствии с приказом Минобрнауки России от 19.12.2013 № 1367, одобрен научно-методическим советом (протокол № 8 от 09.04.2014 г.) и утвержден приказом ректора от 23.04.2014 № 224/10..

Макет обновлён с поправками в части подписей на титульной странице, п.3 добавлена строка для указания часов, проводимых в активной и интерактивной формах обучения, добавлен п. 12.1 Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья (протокол НМС № 6 от 15.04.2015 г.), утвержден приказом ректора.