

	Факультет	Математики, физики и информатики
	Кафедра	Информатики и информационных технологий
	Направление подготовки	02.03.03 Математическое обеспечение и администрирование информационных систем
	Направленность (профиль)	Информационные системы и базы данных
	Технологии программирования	Б1.В.ДВ.7

Министерство образования и науки Российской Федерации
 Федеральное государственное бюджетное образовательное учреждение
 высшего образования
 «Тульский государственный педагогический университет им. Л.Н. Толстого»
 ФГБОУ ВО «ТГПУ им. Л.Н. Толстого»

УТВЕРЖДЕНА

на заседании Ученого совета университета
 протокол № 2 от 11.02.2016 г.

Рабочая программа дисциплины «Технологии программирования»

Трудоемкость: 3 зачетные единицы

Квалификация (степень) выпускника: бакалавр


Форма обучения: очная

Год начала подготовки: 2014

Рассмотрена на заседании кафедры информатики и информационных технологий
 протокол № 03 от «18» ноября 2015 г.

Заведующий кафедрой  А.В. Якушин

Одобрена на заседании Ученого совета факультета
 математики, физики и информатики
 протокол № 5 от 17.12.2015 г.

Декан факультета  Реброва И.Ю.

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы.....	3
2. Место дисциплины в структуре ООП бакалавриата.....	3
3. Объем дисциплины и виды учебной работы	4
4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий	4
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.....	5
6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине.....	6
6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы	6
6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания	6
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.....	7
6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.....	12
7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.....	13
7.1. Основная литература	13
7.2. Дополнительная литература	13
8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	14
9. Методические указания для обучающихся по освоению дисциплины	14
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем	15
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	16
12. Аннотация рабочей программы дисциплины.....	17

1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Достижение планируемых результатов обучения, соотнесенных с общими целями и задачами ОПОП, является целью освоения дисциплины.

Планируемые результаты освоения образовательной программы (код и название компетенции)	Планируемые результаты обучения	Этапы формирования компетенции в процессе освоения образовательной программы
Готовность анализировать проблемы и направления развития технологий программирования (ОПК-3)	<p>Выпускник знает: Основы объектно-ориентированного подхода к программированию.</p> <p>Умеет: Работать с современными системами программирования, включая объектно-ориентированные.</p> <p>Имеет: практические навыки применения библиотек классов при разработке программного обеспечения.</p>	2 этап из 2 (5 семестр)
Владение знаниями о содержании, основных этапах и тенденциях развития программирования, математического обеспечения и информационных технологий» (ПК-7)	<p>Выпускник знает: Основные стандарты в области инфокоммуникационных систем и технологий, в том числе стандарты Единой системы программной документации.</p> <p>Умеет: Ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, разрабатывать основные программные документы.</p> <p>Владеет: практическими навыками применения инструментальных средств при разработке программного обеспечения.</p>	2 этап из 2 (5 семестр)

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП БАКАЛАВРИАТА

Дисциплина «Технологии программирования» относится к дисциплинам по выбору вариативной части образовательной программы. Изучение данной дисциплины осуществляется в 5 семестре. Изучение данной дисциплины базируется на освоении студентами дисциплин «Технологии веб-программирования» и «Объектно-ориентированное программирование».

К началу изучения дисциплины студенты должны владеть:

знаниями

определения и свойства модели, алгоритма и программы, иметь представление о способах записи алгоритмов и реализации основных алгоритмических конструкций следования, ветвления и повторения в любом языке программирования.

умениями

решать типовые задачи по программированию с использованием базовых алгоритмических конструкций и подпрограмм

навыками и (или) опытом деятельности

методами декомпозиции сложных задач на независимые подзадачи

Дисциплина «Технологии программирования» является базовой для дисциплин «Менеджмент проектов», «Информационные системы» и «Программная инженерия».

3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Объем зачетных единиц / часов по формам обучения
Максимальная учебная нагрузка (всего)	108/3
Контактная работа обучающихся с преподавателем (всего)	22
в том числе:	
лекции	8
лабораторные занятия (включая защиту отчета по лабораторным работам)	
семинарские занятия	
практические занятия	12
контрольные работы	
другие виды контактной работы (КСРС)	2
Самостоятельная работа студента (всего)	86
в том числе:	
внеаудиторная самостоятельная работа по подготовке к лекционным занятиям	22
внеаудиторная самостоятельная работа по подготовке к лабораторным занятиям и защите отчета	
внеаудиторная самостоятельная работа при подготовке к семинарским и/или практическим занятиям	30
подготовка учебного проекта	
подготовка к контрольной работе	
выполнение заданий для самостоятельной работы в системе управления обучением MOODLE	30
выполнение курсового проекта (работы)	
подготовка к зачету	4
подготовка к экзамену	
другие виды самостоятельной работы студента	
Промежуточная аттестация в форме зачета	

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ) С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ИЛИ АСТРОНОМИЧЕСКИХ ЧАСОВ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ

Наименование тем (разделов).	Количество академических или астрономических часов по видам учебных занятий			
	Занятия лекционного типа	3 Практические занятия	Другие виды учебных занятий	Самостоятельная работа обучающихся
Тема 1. Организация процесса проектирования программного обеспечения	2	4		20
Тема 2. Технология создания программного кода	2	2		20
Тема 3. . Технологии коллективной разработки программного обеспечения	2	4		22
Тема 4. Методы отладки и тестирования программ.	2	2		20
Контроль самостоятельной работы студентов			2	

Технологии программирования			Б1.В.ДВ.7	
Индивидуальные консультации				
Подготовка к зачету				4
Групповые консультации				
ИТОГО	8	12	2	86
<p>Тема 1. Организация процесса проектирования программного обеспечения Системный подход при разработке ПС. Модели разработки: каскадная, с промежуточным контролем, спиральная и т.д.; CASE и RAD-технологии. Тестирование и оценка качества. Управление проектом, планирование и распределение ресурсов, контроль исполнения сроков. Использование декомпозиции и абстракции при проектировании ПО. Спецификация процедур и данных. Внешняя и внутренняя спецификации. Декомпозиция задачи. Методы проектирования структуры ПО. Методы защиты программ и данных. Жизненный цикл программного средства. Парадигмы программирования: визуальная, функциональная, процедурная, объектно-ориентированная и т.д. Объектно-ориентированная парадигма: понятия объекта, класса объектов; основные понятия объектно-ориентированного программирования (инкапсуляция, наследование и полиморфизм); классы и объекты; интерфейсы и реализация.</p> <p>Тема 2. Технология создания программного кода Библиотеки стандартных компонентов, библиотеки объектов. Проектирование интерфейса с пользователем. Структуры диалога; поддержка пользователя; многооконные интерфейсы; примеры реализации интерфейсов с пользователем с использованием графических пакетов. «Заглушки». «Маленькие хитрости» в программировании. Статические, полустатические и динамические типы данных. Простые и составные типы данных, операция квалификации. Технологии распределенных вычислений: RPC, RMI, Corba, DCOM.</p> <p>Инструментальная среда разработки. Библиотека VCL. Средства поддержки проекта. Отладчики. CASE-технология. UML-диаграммы.</p> <p>Тема 3. . Технологии коллективной разработки программного обеспечения Обзор и классификация средств поддержки коллективной разработки ПО. Программные средства планирования и управления процессом разработки. Сетевые графики и диаграммы рабочего процесса. Сценарии выполнения работ. Применение систем управления документами. CASE-технологии.</p> <p>Тема 4. Методы отладки и тестирования программ. Категории программных ошибок. Типы тестов. Тестирование на этапе планирования. Тестирование на этапе проектирования. Тестирование "белого ящика" на стадии кодирования. Регрессионное тестирование. Тестирование "черного ящика". Разработка тестов.</p> <p>Документация, создаваемая в процессе разработки программных средств. ЕСПД. Пользовательская документация программных средств. Документация по сопровождению программных средств. Стандарт ISO 9126. Модель качества. Характеристики и субхарактеристики качества программного средства. Метрики качества программного средства. Оценивание характеристик качества программных средств.</p> <p style="text-align: center;">5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ</p> <p>Преподавание дисциплины предполагает использование следующего учебно-методического обеспечения.</p> <p>Комплекта мультимедийных презентаций для лекционных занятий.</p> <p>Теоретического курса и информационных приложений, размещенных в электронной образовательной среде MOODLe.</p> <p>Комплекса заданий для практических занятий, размещенных в электронной образовательной среде MOODLe.</p>				
Тула			Страница 5 из 19	

Виды самостоятельной работы обучающихся: выполнение заданий на практические занятия, выполнение индивидуального проектного задания.

При подготовке к занятиям и выполнении самостоятельной работы студентам доступны следующие учебно-методические ресурсы, перечисленные в п.7 рабочей программы, а также электронный учебный ресурс размещенный в среде электронного обучения ТГПУ им. Л.Н. Толстого (<http://moodle.tsput.ru>)

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

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

Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы представлен в таблице пункта 1 рабочей программы.

Формирование компетенции «Готовность анализировать проблемы и направления развития технологий программирования» (ОПК-3) осуществляется в течение двух этапов освоения основной профессиональной образовательной программы.

Первый этап формирования компетенции осуществляется в процессе освоения одной из дисциплин по выбору «Технологии веб-программирования» или «Математические основы исследований».

Второй этап формирования компетенции осуществляется в процессе освоения дисциплины «Компьютерная графика» и одной из дисциплин по выбору «Технологии программирования» или «Экономическая статистика».

Формирование компетенции «Владение знаниями о содержании, основных этапах и тенденциях развития программирования, математического обеспечения и информационных технологий» (ПК-7) осуществляется в течение двух этапов освоения основной профессиональной образовательной программы.

Первый этап формирования компетенции осуществляется в процессе освоения дисциплины «Введение в программирование».

Второй этап формирования компетенции осуществляется в процессе освоения одной из дисциплин «Технологии программирования» или «Экономическая статистика».

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

Дескриптор компетенций	Показатели оценивания	Критерии оценивания
Знания	основных стандартов в области инфокоммуникационных систем и технологий, в том числе стандарты; единой системы программной документации; основы объектно-ориентированного подхода к программированию;	Отметка «зачтено» выставляется, если студент в целом за семестр набрал от 41 до 100 баллов (с учетом баллов, набранных на промежуточной аттестации (зачете)).
Умения	ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, разрабатывать основные программные документы	Отметка «не зачтено» выставляется, если студент в целом за семестр набрал менее 41 балла (с учетом баллов, набранных на промежуточной аттестации (зачете)).
Навыки и опыт деятельности	Работать с современными системами программирования, включая объектно-ориентированные	

Критерии оценивания компетенций формируются на основе балльно-рейтинговой системы с помощью всего комплекса методических материалов, определяющих процедуры оценивания знаний, умений, навыков и опыта деятельности, характеризующих данный этап формирования компетенций.

Баллы, набранные студентом в течение семестра	Баллы за промежуточную аттестацию (зачет)	Общая сумма баллов за модуль в семестр	Отметка
21 – 70	20 – 30	41-100	Зачтено
0 – 20	0 – 20	0 – 40	Не зачтено

Оценка «зачтено» ставится, если студент освоил программный материал всех разделов, последователен в изложении программного материала, достаточно последовательно и логически стройно его излагает, умеет увязывать теорию с практикой, успешно прошел текущий контроль успеваемости по дисциплине, продемонстрировал индивидуальные знания, умениями и навыки практической работы.

Оценка «не зачтено» ставится, если студент не знает значительной части программного материала, допускает существенные ошибки, непоследователен в его изложении, не прошел текущий контроль успеваемости, не в полной мере владеет необходимыми знаниями, умениями и навыками при выполнении практических заданий, то есть студент не может продолжить обучение без дополнительной подготовки по соответствующей дисциплине.

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

Образцы заданий к практическим занятиям:

Пример 1.

Изучить технологию взаимодействия с системой контроля версий.

Дан список задач для разработки программы:

Дан массив из 10 чисел. Сколько элементов массива больше своих «соседей», - т.е. предыдущего и последующего. Первый и последний элементы не рассматривать. (л/р 1)

Для массива из 10 чисел найти номер первого элемента, большего 25.(л/р 1)

В массиве из 10 чисел найти сумму элементов больших, чем второй элемент этого массива.(л/р 1)

Определить-, превосходит ли первый элемент массива из десяти чисел среднее значение элементов этого массива.

Дан массив из 10 чисел. Определить- сколько раз меняется знак у его элементов.

Найти, сколько элементов массива из 10 чисел больше, чем четвертый элемент этого массива.

Найти сумму элементов массива из 10 чисел, меньших, чем 21.

Дан массив из 10 чисел. Увеличить на единицу значения всех элементов кратных 5.

Для массива из 10 целых чисел подсчитать сумму элементов,- значения которых не кратны 3.

Для массива из десяти чисел подсчитать-, сколько чисел меньше первого элемента массива и одновременно больше последнего- элемента.

Дан массив из 10 чисел. Подсчитать- количество- не отрицательных элементов массива.

Дан массив из 10 разных чисел. Найти элемент, меньше всего отличающийся от второго. Указание: функция абсолютной- величины – ABS.

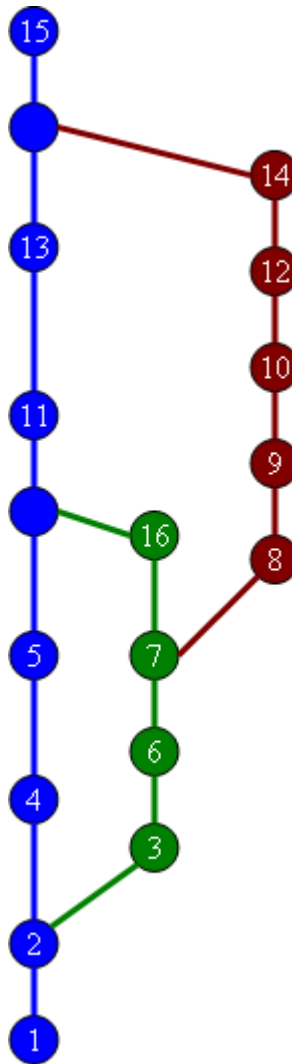
Дан массив из 10 чисел. Подсчитать- количество- не нулевых элементов массива.

В массиве из 10 целых чисел подсчитать- количество- элементов,- кратных 3.

Найти сумму элементов массива из 10 чисел, меньших, чем 5-й элемент этого массива.

Отобразить массив виде гистораммы.

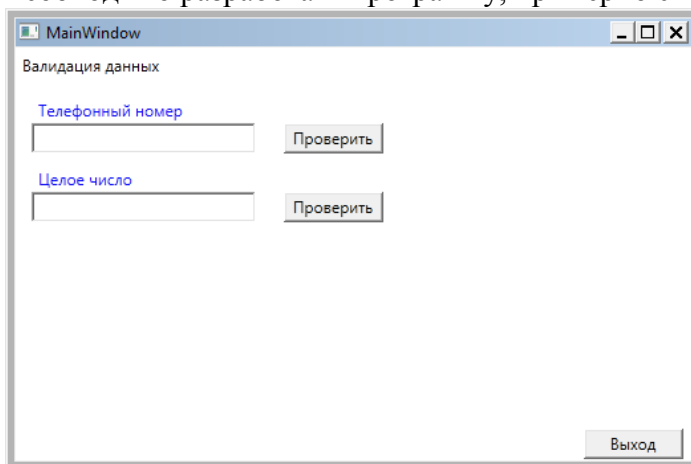
По заданному дереву разработки



Реализовать дерево разработки по указанной схеме. Номер в коммите обозначает задачу.

Пример 2.

Необходимо разработать программу, примерно с таким интерфейсом:



В приложении нужно реализовать валидацию данных.

Составьте регулярное выражение проверяющие, что введенная строка является:

1. Натуральным числом
2. Вещественным числом с 2 знаками после запятой

3. Целым числом
4. Комплексным числом
5. Почтовым индексом
6. Почтовым адресом
7. HTML - контейнером
8. URL
9. Фамилией именем и отчеством
10. Телефонным номером в 11-значном формате, например
 - o 89876543210
 - o +79876543210
 - o 8(987)6543210
 - o +7(987)6543210
 - o 8 (987) 6543210
 - o +7 (987) 6543210
 - o 8 (987) 65-43-210
 - o 8 (987) 65 – 43 – 210
11. Адресом электронной почты
12. Адресом электронной почты в домене ru
13. Действительной календарной датой
14. Без орфографических ошибок написания жи, ши, ча, ща, чу, шу.
15. Целым числом, разряды числа могут разделяться пробелами
16. Удалить из текста все подряд идущие повторы слов. Например:
Эта фраза фраза после обработки обработки обработки должна стать просто как фраза.
-> Эта фраза после обработки должна стать просто как фраза.

Вопросы к зачету

1. Технология программирования и основные этапы ее развития.
2. Жизненный цикл и этапы разработки программного обеспечения.
3. Модели жизненного цикла программного обеспечения.
4. Технология RAD. Ускорение разработки программного обеспечения.
5. Основные качественные и экономические критерии программного обеспечения. Серия стандартов ISO 9000.
6. Проектирование программного обеспечения при структурном подходе.
7. Унифицированный язык моделирования UML. Проектирование классов.
8. Разработка пользовательских интерфейсов.
9. Тестирование программных продуктов.
10. Отладка программного обеспечения.

Индивидуальное проектное задание заключается в разработке программы, удовлетворяющей системе требований:

1. Графический интерфейс пользователя(D)
2. Меню(D)
3. Всплывающие подсказки
4. Ввод данных вручную(D);
5. Ввод данных случайным образом(D);
6. Ввод данных по заданной формуле(D)
7. Ввод данных из файла(D)
8. Реализация заданного численного метода(D)
9. Вычисления с заданной точностью (D)
10. Сохранение данных в текстовый файл(D)

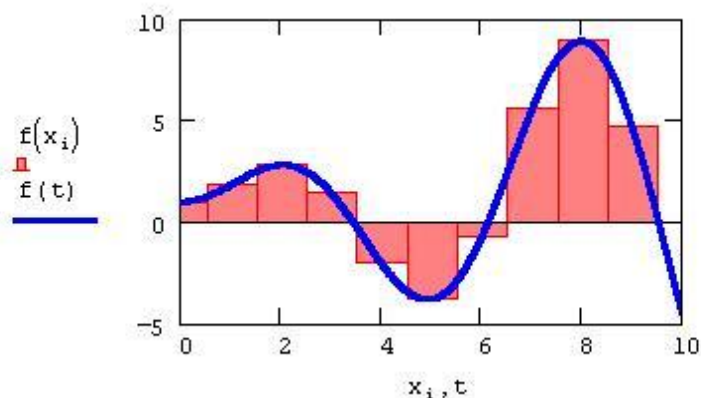
11. Сохранение результатов в word(D);
12. Графическое отображение результатов расчета(D)
13. Окно «О программе»(D)
14. Индикация процесса вычисления
15. Исправление ошибки
16. Обработка исключительных ситуаций
17. Контроль (валидация) входных данных (D)
18. Протоколирование (логирование) действий пользователя
19. Сохранение графического отображения результатов расчета

Буквой D помечены обязательные задания

Примеры индивидуальных проектных заданий:

Метод прямоугольников

Метод прямоугольников — метод численного интегрирования функции одной переменной, заключающийся в замене подынтегральной функции на многочлен нулевой степени, то есть константу, на каждом элементарном отрезке.



Если рассмотреть график подынтегральной функции, то метод будет заключаться в приближённом вычислении площади под графиком суммированием площадей конечного числа прямоугольников, ширина которых будет определяться расстоянием между соответствующими соседними узлами интегрирования, а высота — значением подынтегральной функции в этих узлах. Алгебраический порядок точности равен 0. (Для формулы средних прямоугольников равен 1). При разбиении отрезка на n элементарных отрезков значение интеграла находится по формулам:

Для левых прямоугольников:

Для правых прямоугольников:

Для средних прямоугольников:

$$\int_a^b f(x) dx \approx \sum_{i=0}^{n-1} f\left(\frac{x_i + x_{i+1}}{2}\right) (x_{i+1} - x_i) = \sum_{i=1}^{n-1} f\left(\frac{x_{i-1} + x_i}{2}\right) (x_i - x_{i-1}).$$

Для формул правых и левых прямоугольников погрешность составляет:

$$E(f) = \frac{f'(\xi)}{2} (b - a)^2.$$

Для формулы прямоугольников (средних):

$$E(f) = \frac{f''(\xi)}{24} (b - a)^3.$$

Метод Эйлера для решения ОДУ

Метод Эйлера — наиболее простой численный метод решения (систем) обыкновенных дифференциальных уравнений. Метод Эйлера является явным, одношаговым методом первого порядка точности, основанном на аппроксимации интегральной кривой кусочно-линейной функцией.

Пусть дана задача Коши для уравнения первого порядка: $y|_{x=x_0} = y_0$, где функция f определена на некоторой области $D \subset \mathbb{R}^2$. Решение разыскивается на интервале $(x_0, b]$. На этом интервале введем узлы $x_0 < x_1 < \dots < x_n \leq b$.

Приближенное решение в узлах x_i , которое обозначим через y_i определяется по формуле

$$y_i = y_{i-1} + (x_i - x_{i-1})f(x_{i-1}, y_{i-1}), \quad i = 1, 2, 3, \dots, n.$$

Эти формулы обобщаются на случай систем обыкновенных дифференциальных уравнений.

Метод половинного деления для решения нелинейных уравнений

Метод половинного деления один из методов решения нелинейных уравнений и основан на последовательном сужении интервала, содержащего единственный корень уравнения $F(x)=0$ до того времени, пока не будет достигнута заданная точность ε . Пусть задан отрезок $[a, b]$, содержащий один корень уравнения. Предварительно необходимо определить области локализации корней данного уравнения. Если на отрезке $[a, b]$ содержится более одного корня, то метод не работает.

Алгоритм метода:

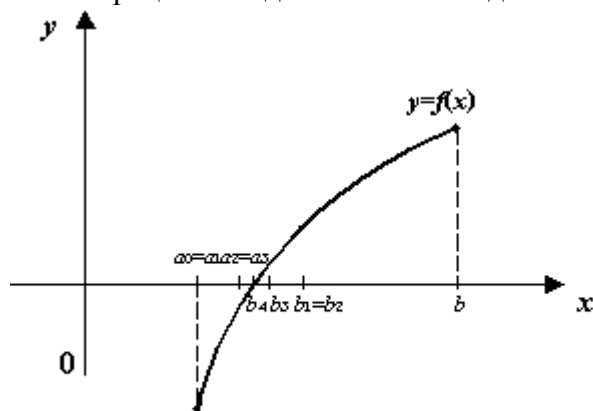
Разобьем отрезок $[a, b]$ пополам. Определим новое приближение корня x в середине отрезка $[a, b]$: $x = (a+b)/2$.

Найдем значения функции в точках a и x : $F(a)$ и $F(x)$. Проверим условие $F(a) \cdot F(x) < 0$.

Если условие выполнено, то корень расположен на отрезке $[a, x]$. В этом случае необходимо точку b переместить в точку x ($b=x$).

Если условие не выполнено, то корень расположен на отрезке $[x, b]$. В этом случае необходимо точку a переместить в точку x ($a=x$). Перейдем к пункту 1 и вновь поделит отрезок пополам. Алгоритм выполнять до тех пор, пока не будет выполнено условие $|F(x)| < \varepsilon$.

Иллюстрация метода половинного деления:



Критерии оценки проектов

Составляющие про-	Критерии для оценивания	Максимальное
Тула		Страница 11 из 19

Технологии программирования		Б1.В.ДВ.7
екта		количество баллов
Постановка проблемы и ее обоснованность, формулирование целей и задач	<ul style="list-style-type: none"> общественная значимость и актуальность выдвинутых проблем; соответствие темы, цели и задач проекта; разумность масштаба работ. 	10
Содержание проекта/ проектной разработки	<ul style="list-style-type: none"> логичность, взаимосвязь и последовательность этапов проекта; адекватность предлагаемых мероприятий решению поставленных задач; корректность используемых методов работы; четкость определения целевой группы и обоснованность её участия при реализации проекта; соответствие теоретической, эмпирической и проектной частей, их связь с практикой и выбранным видом профессиональной деятельности; соблюдение заявленных временных рамок реализации проекта; самостоятельность и активность участника проекта. 	10
Результат выполнения прикладного проекта	<ul style="list-style-type: none"> соответствие ожиданий от проекта / планируемого результата полученному продукту; степень решения заявленной проблемы; успешность преодоления трудностей в реализации проекта; оценка участников целевой группы; перспективы развития проекта после завершения проекта; возможность тиражирования проекта. 	10
Презентация результатов работы над прикладным проектом	<ul style="list-style-type: none"> ясность, логичность, профессионализм изложения доклада; наглядность и структурированность материала презентации; умение корректно использовать профессиональную лексику и понятийно-категориальный аппарат. 	10
Ответы на вопросы	<ul style="list-style-type: none"> степень владения темой; ясность аргументации взглядов студента, презентующего результаты выполнения проекта; четкость и лаконичность ответов на вопросы. 	10

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

Описание балльно-рейтинговой системы по дисциплине.

Итоговая рейтинговая оценка по дисциплине складывается из следующих составляющих:

- 1) В течение семестра за выполнение заданий по курсу студент может максимально получить 70 баллов;
- 2) Обязательной формой текущей аттестации знаний является выполнение индивидуального проектного задания 20 баллов.

3) На зачёте ответ студента может быть максимально оценен в 30 баллов.

При этом, для получения положительной итоговой оценки на зачете необходимо получить не менее 60% по каждой составляющей и выполнить все задания для практических занятий. Шкала перевода баллов в оценку: до 41 - «не зачтено»; 41 - 100 - «зачтено».

№ п/п	Критерии оценивания	Максимальное количество баллов	Баллы, полученные студентом
1.	Выполнение заданий:	60	
1.1.	Практические занятия	40	
1.2.	Индивидуальное проектное задание	20	
3.	Зачет	40	
	ИТОГО:	100	

7. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

7.1. Основная литература

1. Золотов, С. Ю. Проектирование информационных систем [Электронный ресурс] : учебное пособие / С. Ю. Золотов. - Томск : Эль Контент, 2013. - 88 с. - 978-5-4332-0083-8 : Б. ц.
URL: <http://biblioclub.ru/index.php?page=book&id=208706>
2. Москвитин, А. А. Решение задач на компьютерах: Ч. 1. Постановка (спецификация) задач : учебное пособие / А. А. Москвитин. - М. : Директ- Медиа, 2015. - 185 с. - ISBN 978-5-4475-3651-0 : Б. ц.
URL: http://biblioclub.ru/index.php?page=book_view&book_id=273666

7.2. Дополнительная литература

1. Основы информатики и начала программирования [Текст] : учебное пособие / В. Г. Куперман, Е. Г. Торина, Изд. 2-е, перераб. и доп. - Тула : Изд-во ТГПУ им. Л. Н. Толстого, 1997. - 264 с. - ISBN 5879541169
2. Технология программирования [Текст] : учебник для студентов вузов / Г. С. Иванова, 2-е изд., стер. - М. : Изд-во МГТУ им. Н. Баумана, 2003. - 320 с. - ISBN 5703820774
3. Информационные системы [Текст] : учебное пособие для вузов / Ю. С. Избачков, В. Н. Петров. - 2-е изд. - СПб. : Питер, 2005. - 655 с. - ISBN 978-5-469-00641-1 : Б. ц.
URL: <http://ibooks.ru/reading.php?productid=21650>
4. Технология программирования на С++ [Текст] : учебное пособие для вузов / Н. А. Литвиненко. - СПб. : БХВ-Петербург, 2005. - 288 с. - ISBN 5-94157-655-2
5. Электронные устройства информационных систем и автоматики [Текст] : учебник / Э. В. Ромаш, В. В. Ефремов, Н. А. Феоктистов. - М. : [б. и.], 2012. - 248 с. - ISBN 978-5-394-01701-8 : Б. ц.
URL: http://biblioclub.ru/index.php?page=book_view&book_id=115786

8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1. ИКТ [Электронный ресурс] : федеральный образовательный портал / ФГАУ ГНИИ ИТТ "Информатика". - М. : [б. и.], 2003. - Загл. с титул. экрана.
URL: <http://www.ict.edu.ru>
2. CITForum.ru [Электронный ресурс] : образовательный портал / "ЦИТ Форум". - [Б. м. : б. и.], 1997. - Загл. с титул. экрана.
URL: <http://citforum.ru/>
3. Виртуальный компьютерный музей [Электронный ресурс] : сайт / Э. Пройдаков. - М. : [б. и.], 1997. - Загл. с титул. экрана. - Б. ц.
URL: <http://www.computer-museum.ru>

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Приступая к изучению новой учебной дисциплины, студенты должны ознакомиться с учебной программой, учебной, научной и методической литературой, имеющейся в библиотеке университета, встретиться с преподавателем, ведущим дисциплину, получить в библиотеке рекомендованные учебники и учебно-методические пособия, осуществить запись на соответствующий курс в среде электронного обучения университета.

Глубина усвоения дисциплины зависит от активной и систематической работы студента на лекциях и практических занятиях, а также в ходе самостоятельной работы, по изучению рекомендованной литературы.

На лекциях важно сосредоточить внимание на ее содержании. Это поможет лучше воспринимать учебный материал и уяснить взаимосвязь проблем по всей дисциплине. Основное содержание лекции целесообразнее записывать в тетради в виде ключевых фраз, понятий, тезисов, обобщений, схем, опорных выводов. Необходимо обращать внимание на термины, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставлять в конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющей материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. С целью уяснения теоретических положений, разрешения спорных ситуаций необходимо задавать преподавателю уточняющие вопросы. Для закрепления содержания лекции в памяти, необходимо во время самостоятельной работы внимательно прочесть свой конспект и дополнить его записями из учебников и рекомендованной литературы. Конспектирование читаемых лекций и их последующая доработка способствует более глубокому усвоению знаний, и поэтому являются важной формой учебной деятельности студентов.

Прочное усвоение и долговременное закрепление учебного материала невозможно без продуманной самостоятельной работы. Такая работа требует от студента значительных усилий, творчества и высокой организованности. В ходе самостоятельной работы студенты выполняют следующие задачи: дорабатывают лекции, изучают рекомендованную литературу, готовятся к практическим занятиям, к коллоквиуму, контрольным работам по отдельным темам дисциплины. При этом эффективность учебной деятельности студента во многом зависит от того, как он распорядился выделенным для самостоятельной работы бюджетом времени.

Результатом самостоятельной работы является прочное усвоение материалов по предмету согласно программы дисциплины. В итоге этой работы формируются профессиональные умения и компетенции, развивается творческий подход к решению возникших в ходе учебной деятельности проблемных задач, появляется самостоятельности мышления.

Целью практических занятий по данной дисциплине является закрепление теоретических знаний, полученных при изучении дисциплины.

При подготовке к практическому занятию целесообразно выполнить следующие рекомендации: изучить основную литературу; ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях: журналах, газетах и т. д.; при необходимости доработать конспект лекций. При этом учесть рекомендации преподавателя и требования учебной программы.

При выполнении практических занятий основным методом обучения является самостоятельная работа студента под управлением преподавателя. На них пополняются теоретические знания студентов, их умение творчески мыслить, анализировать, обобщать изученный материал, проверяется отношение студентов к будущей профессиональной деятельности.

Оценка выполненной работы осуществляется преподавателем комплексно: по результатам выполнения заданий, устному сообщению и оформлению работы. После подведения итогов занятия студент обязан устранить недостатки, отмеченные преподавателем при оценке его работы.

Преподавание дисциплины должно включать в себя следующие образовательные технологии:

- 1) Проведение лекций с использованием презентаций на основе мультимедийных технологий;
- 2) Обеспечение студентов сопутствующими материалами, размещенными в среде Moodle; Описание практических занятий по дисциплине

Полные варианты практических занятий размещены в системе управления обучением MOODLE.

№	Наименование практических занятий	Объем в часах
1	Создание тестового приложения	2
2	Настройка связи VS и Git	2
3	Создание дерева разработки	4
4	Валидация данных.	4
	Итого	12

Типовые задания для самостоятельной работы по дисциплине

1. Дан массив из n чисел. Сколько элементов массива больше своих «соседей», - т.е. предыдущего и последующего. Первый и последний элементы не рассматривать.
2. Для массива из n чисел найти номер первого элемента, большего 25.
3. В массиве из n чисел найти сумму элементов больших, чем второй элемент этого массива.

10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ

1. Специально оборудованные аудитории и компьютерные классы: персональные компьютеры (модели: Intel Pentium4, AMD Athlon, AMD Duron), мультимедийные проекторы, аудиовизуальные устройства;
2. Программное обеспечение в соответствии с программой курса;
3. Методические пособия и литература в библиотеке университета и на кафедре.
4. Студентам обеспечен доступ к сети Internet.

Перечень лицензионного программного обеспечения, используемого при освоении дисциплины:

1. Подписка Microsoft DreamSpark Premium - Сублицензионный договор № S-2042626/M18 от 04.06.2013:

- 1.1. Средства для разработки и проектирования Visual Studio 2008, 2010, 2012 и 2013 Professional Editions;
 - 1.2. Операционная система Windows 7 Professional;
 - 1.3. Операционная система Windows 8 Pro;
 - 1.4. Операционная система Windows 8.1 Pro;
 - 1.5. Отдельные программы из Office 2007, Office 2010, Office 2013 (в том числе Access, Visio, Project и др.);
2. Свободное программное обеспечение по лицензии GNU
- 2.1. Debian Linux Weezy
 - 2.2. Apache Web Server
 - 2.3. MySQL
 - 2.4. PHP 5.0
 - 2.5. Domain Technologie Control Контрольная панель локального хостинга
- У обучающихся имеется доступ (удаленный доступ), в том числе в случае применения электронного обучения, дистанционных образовательных технологий, к современным профессиональным базам данных и информационным справочным системам, состав которых определяется в рабочих программах дисциплин и подлежит ежегодному обновлению:
1. Компьютерная информационно-правовая система «Гарант» - регистрационный номер клиента 71-70685-000033.
 2. Официальный интернет-портал правовой информации <http://pravo.gov.ru>.
 3. Портал Федеральных государственных образовательных стандартов высшего образования <http://fgosvo.ru>.
 4. Math-Net.Ru [Электронный ресурс] : общероссийский математический портал / Математический институт им. В. А. Стеклова РАН ; Российская академия наук, Отделение математических наук. - М. : [б. и.], 2010. - Загл. с титул. экрана. - Б. ц. URL: <http://www.mathnet.ru>
 5. ИКТ [Электронный ресурс] : федеральный образовательный портал / ФГАУ ГНИИ ИТТ "Информика". - М. : [б. и.], 2003. - Загл. с титул. экрана. - Б. ц. URL: <http://www.ict.edu.ru>
 6. Университетская библиотека Online [Электронный ресурс] : электронная библиотечная система / ООО "Директ-Медиа" . - М. : [б. и.], 2001. - Загл. с титул. экрана. - Б. ц. URL: www.biblioclub.ru
 7. Универсальные базы данных East View [Электронный ресурс] : информационный ресурс / East View Information Services. - М. : [б. и.], 2012. - Загл. с титул. экрана. - Б. ц. URL: www.ebiblioteka.ru
 8. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс] : информационный портал / ООО "РУНЭБ" ; Санкт-Петербургский государственный университет. - М. : [б. и.], 2005. - Загл. с титул. экрана. - Б. ц. URL: www.eLibrary.ru

11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

1. Учебные аудитории для проведения занятий лекционного типа, оборудованные мультимедийными средствами обучения.
2. Учебные аудитории для проведения практических занятий.
3. Компьютерные классы с доступом в интернет для работы с информационно-правовыми системами, в том числе «Гарант» и с доступом к электронно-библиотечной системе.
4. Аудитории для самостоятельной работы студентов, оснащенные компьютерной техникой, имеющей доступ к информационно-телекоммуникационной сети «Интернет», электронной информационно-образовательной среде ТГПУ им. Л.Н. Толстого, внутривузовскому сетевому окружению.

12. АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ.

1. Планируемые результаты обучения при освоении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы.

В результате освоения дисциплины у студента должна быть сформирована следующая компетенция: «Готовность анализировать проблемы и направления развития технологий программирования» (ОПК-3).

В результате освоения дисциплины студент должен приобрести:

знания основы объектно-ориентированного подхода к программированию;

умения работать с современными системами программирования, включая объектно-ориентированные.

навыки применения библиотек классов при разработке программного обеспечения.

В результате освоения дисциплины у студента должна быть сформирована следующая компетенция: «Владение знаниями о содержании, основных этапах и тенденциях развития программирования, математического обеспечения и информационных технологий» (ПК-7).

В результате освоения дисциплины студент должен приобрести:

знания основных стандартов в области инфокоммуникационных систем и технологий, в том числе стандарты; единой системы программной документации;

умения ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, разрабатывать основные программные документы.

навыки применения инструментальных средств при разработке программного обеспечения.

2. Место дисциплины в структуре ОПОП.

Дисциплина «Технологии программирования» относится к дисциплинам по выбору вариативной части образовательной программы. Изучение данной дисциплины осуществляется в 5 семестре. Изучение данной дисциплины базируется на освоении студентами дисциплин «Технологии веб-программирования» и «Объектно-ориентированное программирование».

К началу изучения дисциплины студенты должны владеть:

знаниями

определения и свойства модели, алгоритма и программы, иметь представление о способах записи алгоритмов и реализации основных алгоритмических конструкций следования, ветвления и повторения в любом языке программирования.

умениями

решать типовые задачи по программированию с использованием базовых алгоритмических конструкций и подпрограмм

навыками и (или) опытом деятельности

методами декомпозиции сложных задач на независимые подзадачи

Дисциплина «Технологии программирования» является базовой для дисциплин «Менеджмент проектов», «Информационные системы» и «Программная инженерия».

3. Объем дисциплины: 3 зачетные единицы.

4. Образовательный процесс осуществляется на русском языке.

5. Разработчик: Якушин А.В., к.п.н., доцент, зав. кафедрой И и ИТ.

**13. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ К РАБОЧЕЙ ПРОГРАММЕ
ДИСЦИПЛИНЫ**

- 1) Внесены изменения в п.7 «Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины».
- 2) Обновлен п.10 «Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения и информационных справочных систем» на основании действующих лицензионных соглашений

Заведующий кафедрой ИиИТ



А.В. Якушин

«29» августа 2016 г..

Программа составлена в соответствии с требованиями ФГОС ВО.

Разработчик (и):

Фамилия, имя, отчество	Учёная степень	Учёное звание	Должность	Дата разработки	Подпись
Якушин Алексей Валериевич	к.п.н.	Доц.	зав. кафедры информатики и информационных технологий	17.11.2015	