



Факультет	Математики, физики и информатики	
Кафедра	Информатики информационных технологий	
Направление подготовки	02.03.02 Фундаментальная Информатика и информационные технологии	
Профиль	Открытые информационные системы	
Языки программирования		Б1.Б.17

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 февраля 2016 г.


Учебная программа дисциплины «Языки программирования»

Трудоемкость: 5 зачетных единиц

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Рассмотрена на заседании кафедры информатики и информационных технологий
протокол № 3 от 18 ноября 2015 г.

Заведующий кафедрой  А.В.Якушин

Одобрена на заседании Ученого совета факультета
математики, физики и информатики
протокол № 5 от «17» декабря 2015 г.

Декан  И.Ю. Реброва

СОДЕРЖАНИЕ

1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ.....	3
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП.....	3
3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ.....	4
4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ЧАСОВ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ.....	4
5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ».....	5
6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ	5
6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.....	5
6.2. Описание показателей, критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания	5
6.3. Типовые контрольные задания и иные материалы, характеризующие этапы формирования компетенций в процессе освоения образовательной программы.....	7
6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и/или опыта деятельности, характеризующие этапы формирования компетенций. 9	
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	13
9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ .	13
10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ.....	14
11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»	Ошибка! Закладка не определена.
12. АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»	16
13. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»	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 (модуля).

Планируемые результаты освоения образовательной программы (код и название компетенции)	Планируемые результаты обучения	Этапы формирования компетенции в процессе освоения образовательной программы
Способностью решать задачи профессиональной деятельности в составе научно-исследовательского и производственного коллектива (ПК-4)	<p>Выпускник знает:</p> <ul style="list-style-type: none"> • основные алгоритмические конструкции; • способы представления структур данных; • классификации языков программирования; • основные требования к языкам программирования, способы реализации языков; • основные алгоритмы решения задач по программированию; <p>Умеет:</p> <ul style="list-style-type: none"> • выбирать язык и метод программирования в зависимости от решаемой задачи; • разрабатывать алгоритм и программу в соответствии с этапами решения задачи и стандартом языка; • анализировать и доказывать правильность программы и эффективность алгоритма; • грамотно составлять сопроводительную документацию по результатам решения задач; <p>Владеет:</p> <ul style="list-style-type: none"> • навыками квалифицированного применения в профессиональной деятельности современных языков и парадигм программирования; • технологией разработки алгоритмических и программных решений в области системного и прикладного программирования. 	1 этап из 4 (4 семестр)

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Языки программирования» относится к блоку обязательных дисциплин вариативной части дисциплин направления.

К началу освоения дисциплины студенты по результатам изучения предметов «Основы алгоритмизации» и «Вводный курс программирования» должны владеть:

- знаниями определения и свойства модели, алгоритма и программы;
- умениями записи алгоритмов и реализации основных алгоритмических конструкций следования, ветвления и повторения в любом языке программирования или на псевдокоде;
- навыками и опытом деятельности поиска информации и реферирования профессиональной литературы.

Дисциплина «Языки программирования» является базовой для качественного изучения основ и методов объектно-ориентированного программирования, основ и принципов объектно-ориентированного и логического программирования, выполнения выпускной квалификационной работы.

3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Объем часов/ зачетных единиц по формам обучения
	очная
Максимальная учебная нагрузка (всего)	180/5
Контактная работа обучающихся с преподавателем (всего)	72
в том числе:	
лекции с применением мультимедийных технологий и раздаточным материалом для студентов	28
лабораторные занятия с использованием современных информационных технологий по разработке алгоритмов и программ	42
контрольные работы	2
Самостоятельная работа студента (всего)	72
в том числе:	
внеаудиторная самостоятельная работа при подготовке к лабораторным и практическим занятиям	42
подготовка к контрольной работе	4
Выполнение заданий для самостоятельной работы в модульной объектно-ориентированной динамической учебной среде Moodle	26
Подготовка к экзамену	36
<i>Промежуточная аттестация в форме: экзамена</i>	

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

Наименование тем (разделов).	Количество академических или астрономических часов по видам учебных занятий			
	Занятия лекционного типа	Занятия лабораторного типа	Другие виды учебных занятий	Самостоятельная работа обучающихся
Тема 1.1. Введение	4	8		10
Тема 1.2. Обработка исключений	6	8		8
Тема 1.3. Введение в библиотеку STL	6	8		10
Тема 1.4. Контейнеры и итераторы	6	8		10
Тема 1.5. Функциональные объекты	6	10		14
Контроль самостоятельной работы студентов			2	
Подготовка к экзамену				36
г. Тула				Стр. 4 из 18

Групповые консультации				20
ИТОГО	28	42	2	108

Тема 1.1. Введение. Этапы решения задач на ЭВМ. Модель. Алгоритм. Способы записи алгоритма.

Тема 1.2. Обработка исключений. Исключения. Перехват событий. Работа с ошибками во время исполнения программы. Трассировка. Отслеживание и контроль ошибок компилятором. Контроль ошибок со стороны программиста. Назначение и принципы работы с заголовочными файлами. Работа на удаленном рабочем столе. Принципы работы в среде программирования. Средства и методы создания и редактирования решений. Средства отладки программ.

Тема 1.3. Введение в библиотеку STL. Работа с данными с помощью абстрактных библиотек. Работа с вложенными типами данных. Работа со строками. Фактические и формальные параметры функций пользователя. Области действия и видимости переменных. Библиотека функций. Библиотека функций пользователя. Разработка собственных типов данных. Перегрузка типов данных.

Тема 1.4. Контейнеры и итераторы. Обработка сложных типов данных. Инстансы. Экземпляры. Объекты. Перебор инстансов. Сборка мусора. Операторы break, continue, return.

Тема 1.5. Функциональные объекты. Работа с объектами в языке Си. Вложенные функции. Процедурный и объектно-ориентированный подход в работе с итераторами.

5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ «ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

Преподавание дисциплины предполагает использование следующего учебно-методического обеспечения.

Комплекта мультимедийных презентаций для лекционных занятий.

Теоретического курса и информационных приложений, размещенных в электронной образовательной среде MOODLe.

Комплекса заданий для лабораторных работ, размещенных в электронной образовательной среде MOODLe.

Виды самостоятельной работы обучающихся: выполнение заданий на лабораторные работы, 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 «Способностью решать задачи профессиональной деятельности в составе научно-исследовательского и производственного коллектива» (ПК-4) осуществляется в течение четырех этапов освоения основной профессиональной образовательной программы.

Первый этап формирования компетенции осуществляется в процессе освоения дисциплины «Языки программирования».

Второй этап формирования компетенции осуществляется в процессе освоения дисциплины «Теория вероятностей и математическая статистика».

Третий этап формирования компетенции осуществляется в процессе освоения производственной практики по получению профессиональных умений и опыта профессиональной деятельности.

Четвертый этап формирования компетенции осуществляется в процессе освоения дисциплины «Программная инженерия».

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

Дескриптор компетенций	Показатели оценивания	Критерии оценивания
Знания	типы данных, алгоритмические конструкции ЯП; способы описания типов данных и функций пользователя в ЯП; теоретические основы работы с файлами в ЯП; интерфейс среды программирования VS 2010	Отметка «зачтено» выставляется, если студент в целом за семестр набрал от 41 до 100 баллов (с учетом баллов, набранных на промежуточной аттестации (зачете)).
Умения	применять полученные знания при решении практических задач профессиональной деятельности; решать типовые задачи по программированию с использованием базовых алгоритмических конструкций и подпрограмм; осуществлять поиск необходимой информации в глобальных компьютерных сетях и LMS MOODLE.	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	декомпозиции сложных задач на независимые подзадачи; отладки и тестирования программ.	

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
11 – 80	0 – 20	41-100	Зачтено
0 – 20	0 – 20	0 – 40	Не зачтено

Знания, умения, навыки и компетенции студентов по дисциплине оцениваются по двухбалльной шкале с отметками: «зачтено»; «не зачтено». Как правило, при двухбалльной системе преподавателями используются следующие показатели, при условии успешного прохождения текущего контроля успеваемости по дисциплине:

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 и иные материалы, характеризующие этапы формирования компетенций в процессе освоения образовательной программы

Задания, направленные на формирование подходов программирования с использованием объектно-ориентированного подхода.

1. Назовите принципы объектно-ориентированного представления программных систем
2. Какие функции есть у любого класса?
3. Что такое деструктор?
4. Какое значение должен возвращать деструктор?
5. Какая функция выполняет начальную инициализацию данных в классе?
6. Что такое класс?
7. Преобразование десятичного числа в шестнадцатеричное.
8. Составьте математическую модель задачи о распознавания простого числа.
9. Найти в последовательности чисел первое число больше A

Вопросы к зачету

- 1 Язык программирования Си. Общая характеристика языка: место в классификации языков, синтаксис и семантика языка, основные объекты языка, операторы, структура программы.
- 2 Данные числовых типов в языке Си: объявление, характеристика, допустимые операции, приведение типов. Пример использования.
- 3 Операции языка Си. Приоритет операций. Оператор и операция присваивания в языке Си. Множественное присваивание. Выражения.
- 4 Функции форматного ввода и вывода. Параметры. Управляющая строка. Спецификаторы формата. Управляющие символы.
- 5 Алгоритмическая конструкция ветвления: полная и неполная форма, блок-схемы. Условная операция. Условный оператор в языке Си: структура оператора, полная и неполная формы, использование сложных условий. Пример на языке Си.
- 6 Алгоритмическая конструкция выбора: понятие, блок-схема. Оператор выбора в языке Си: структура оператора. Пример программы на Си.
- 7 Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла `for`: структура оператора, пример использования.
- 8 Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла с предусловием `while` в языке Си: структура оператора, допустимые и недопустимые условия, пример использования.
- 9 Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с постусловием (понятие, использование, блок-схема). Оператор цикла с постусловием в языке Си: структура оператора, допустимые и недопустимые условия, пример использования.
- 10 Типовые циклические алгоритмы: максимум/минимум, сумма/произведение, количество.
- 11 Линейный массив: понятие массива, объявление, инициализация массива, индексация элементов. Формирование и вывод массива.

- 12 Типовые алгоритмы для работы с линейными массивами.
- 13 Задача сортировки массива. Алгоритм сортировки линейного массива методом "пузырька".
- 14 Двумерный массив. Объявление, инициализация двумерного массива, индексация элементов. Формирование и вывод двумерного массива.
- 15 Типовые алгоритмы для обработки двумерного массива (целиком).
- 16 Частичная обработка двумерного массива (по строкам или по столбцам). Типовые алгоритмы.
- 17 Указатели. Понятие указателя, объявление, инициализация. Операции, применимые к указателям. Указатель на указатель.
- 18 Использование указателей при работе с массивами и матрицами. Получение адресов и значений элементов. Последовательный перебор элементов. Примеры.
- 19 Функции в языке Си: понятие, объявление, прототипы функций. Вызов функции. Типы возвращаемых значений.
- 20 Функции в языке Си: параметры формальные и фактические, механизм передачи параметров. Передача параметров «по значению» и «по ссылке». Пример использования.
- 21 Локальные и глобальные переменные. Области действия и области видимости. Экранирование переменных.
- 22 Библиотечные функции. Заголовочные файлы. Подключение библиотек. Функции математической библиотеки.
- 23 Рекурсивное описание функций: база рекурсии, рекурсивный вызов, использование стека. Пример использования.
- 24 Динамическая память: выделение и освобождение памяти, размещение данных в динамической памяти. Выделение и освобождение памяти при работе с одиночными переменными и с массивами, изменение размера массива.
- 25 Динамическая память: выделение и освобождение памяти, размещение данных в динамической памяти. Выделение и освобождение памяти при работе с матрицами. Особенности обработки матриц при работе с динамической памятью.
- 26 Массив символов и строка в языке Си. Ввод и вывод строк. Простейшие алгоритмы сканирования и обработки строки. Пример.
- 27 Строка в языке Си: библиотечные функции для обработки строк. Примеры использования.

Примеры типовых заданий для написания программ

Студенту требуется выбрать задание в зависимости от уровня сложности.

Уровень D

1. Функция **add** для суммирования одномерных массивов типа **int** и **float**.
2. Функции **input** и **output** для ввода и вывода на экран массивов и матриц.
3. Функция, которая меняет местами минимальный и максимальный элементы массива.
4. Функция, возвращающая минимальный из трех передаваемых в нее параметров любого (но одинакового) типа.
5. Функция, находящая минимальный элемент в массиве и заменяющая его заданным числом.
6. Функция, меняющей местами значения двух переданных в нее параметров.
7. Функция, подсчитывающая сумму положительных элементов в массиве.

Уровень C

1. Функция **min_el** поиска минимального элемента в массивах и матрицах различного типа.
2. Функция **null_count** определения количества нулей в массивах и матрицах различного типа.
3. Функция **movement** определения пройденного расстояния при равномерном и равноуско-

ренном движениях тела.

4. Функция **distance** вычисления расстояния между двумя точками в двумерном и трехмерном пространстве.
5. Функция, вычисляющая произведение отрицательных элементов массива.
6. Функция, находящая среднее арифметическое элементов массива.

Уровень В

1. Функция **area**, вычисляющая площадь круга по его радиусу, прямоугольника и треугольника по их сторонам.
2. Функция **perimetr**, вычисляющая периметр круга по его радиусу, прямоугольника и треугольника по их сторонам.
3. Функция, формирующая массив, каждый элемент которого равен минимальному из соответствующих значений двух других массивов – параметров.
4. Функция сортировки элементов массива методом пузырька.
5. Функции: **generate** заполнения массивов и матриц случайными числами заданного диапазона и **output** вывода их на экран.
6. Функция, которая считает сумму элементов главной диагонали матрицы.
7. Функция **triangle** вычисления площади треугольников: равностороннего (по стороне), равнобедренного (по основанию и высоте), обычного (по трем сторонам).
8. Функция **volume** определения объемов: шара по его радиусу ($V=4/3\pi R^3$), конуса по радиусу основания и высоте ($V=1/3\pi r^2 h$), усеченного конуса по радиусам двух оснований и высоте ($V=1/3\pi h(r^2+r_1^2+rr_1)$).

Уровень А

1. Функция, формирующая новый массив, упорядоченный по неубыванию, из двух массивов-параметров, уже упорядоченных по неубыванию.
2. Функция **surface_area** определения площади поверхности: шара по его радиусу ($S=4\pi R^2$), цилиндра по радиусу основания и высоте ($S=2\pi r(r+h)$), усеченного конуса по радиусам двух оснований и образующей ($S=\pi((r+r_1)l+r^2+r_1^2)$).
3. Функция, находящая среднееквадратическое значение элементов матрицы.
4. Функция **mult** умножения массива и матрицы на число.
5. Функция сложения двух матриц.
6. Функция **sub_array** нахождения разности двух массивов и матриц.
7. Функция, подсчитывающая количество строк матрицы, сумма элементов которых больше некоторого заданного числа (это число имеет тот же тип, что и элементы матрицы).

6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и/или опыта деятельности, характеризующие этапы формирования компетенций

Рейтинг по дисциплине «Языки программирования»

Составляющие итоговой оценки за дисциплину:

1) Текущий контроль (общий вес 70 баллов):

до 21 балла – посещение занятий;

до 36 баллов – выполнение заданий в ходе выполнения лабораторных работ в LMS Moodle и заданий для самостоятельной работы

до 10 баллов – выполнение отдельно выделенных в методических указаниях к выполнению работ задач повышенной сложности

до 3 баллов – выполнение контрольной работы

Оценка	«зачтено»	«не зачтено»

Интервал количества баллов	41..100	0..40
-------------------------------	---------	-------

2) Итоговый контроль заключается в проведении зачета (общий вес - 30 баллов). Зачет проводится по вопросам с обязательным решением задач. Как правило, студент получает два вопроса из приведенного выше списка и одну задачу, готовится в присутствии преподавателя и дает подробные комментарии. Студент, пропускавший занятия в ходе семестра, получает дополнительные вопросы и задачи по каждой пропущенной им теме (на усмотрение преподавателя).

Для получения положительной итоговой оценки на зачете необходимо получить не менее 50% по каждой составляющей и выполнить все лабораторные работы. Шкала перевода баллов в оценку: до 40 - «не зачтено»; 41 - 100 - «зачтено».

№ п/п	Содержание занятия	количество часов	баллы
1.	Введение	6	3
2.	Тема 1.1. Введение	16	6
3.	Тема 1.2. Обработка исключений	20	20
4.	Тема 1.3. Введение в библиотеку STL	34	30
5.	Тема 1.4. Контейнеры и итераторы	20	8
6.	Контрольная работа	2	3
7.	Зачет	10	30
ИТОГО		108	100

Способность применять в профессиональной деятельности основные методы и средства автоматизации проектирования, производства, испытаний и оценки качества программного обеспечения (ОПК 4)

Планируемые результаты обучения	Критерии оценивания с весовым коэффициентом	Показатели оценивания				
		1	2	3	4	5
Выпускник знает типы данных, алгоритмические конструкции языка программирования C(C++);	когнитивный – 0.1	Знает о существовании типов данных и алгоритмических конструкций языка программирования C(C++) как сущности любого языка программирования	Знает о принципах и способах введения в программу типов данных и алгоритмических конструкций языка	Знает диапазоны значений и множество операций для стандартных типов данных; знает блок-схемы, синтаксис и семантику операторов реализации базовых алгоритмических конструкций языка	Знает семантику команд и сущность процессов, происходящих в среде при введении в программу объектов различных типов данных и алгоритмических конструкций языка	Знает тонкости процессов, происходящих в среде при введении в программу объектов различных типов данных и исполнения алгоритмических конструкций языка
Выпускник знает способы описания типов данных и функ-	когнитивный – 0,1	Может перечислить служебные слова для введения в программу типов данных	Правильно осуществляет ввод в программу данных различных стан-	Знает о способах описания данных как константных и переменных	Знает о сущности «область видимости» для локальных и глобальных	Знает тонкости ввода данных через макросы и директивы препроцессо-

ций пользо- вателя в ЯП;		и сформули- ровать необ- ходимость работы с функциями пользователя	дартных ти- пов; знает возможности языка для описания и реализации функций пользователя	значениях указанного типа; пара- метризации функций	данных, формальных и фактиче- ских пара- метрах функции	ра, знает алго- ритмы органи- зации данных при использо- вании библио- тек функций пользователя
Выпускник знает теоретиче- ские основы работы с файлами в ЯП;	когнитивный – 0,1	Знает основ- ные понятия: файл, режим доступа	Знает поня- тие поток и устройства ввода и вы- вода	Знает типы данных и функции по работе с файлами	Знает функ- ции обмена данными между опе- ративной памятью и внешними файлами	Осуществляет осознанный выбор функ- ций обмена данными меж- ду оператив- ной памятью и внешними файлами
Выпускник умеет: применять полученные знания при решении практиче- ских задач профессио- нальной дея- тельности;	деятельност- ный – 0,1	Способен понять пра- вильно напи- санные про- граммы при решении практических задач проф- фессиональ- ной деятель- ности	Способен понять и объ- яснить ос- новные мо- менты алго- ритма пра- вильно напи- санных про- грамм при решении практических задач проф- ессиональ- ной деятель- ности	Способен модифици- ровать алго- ритм пра- вильно на- писанной программы при несуще- ственном изменении условия практиче- ских задач профессио- нальной дея- тельности	Способен самостоя- тельно напи- сать алго- ритм и про- грамму при решении практиче- ских задач профессио- нальной дея- тельности	Способен са- мостоятельно написать алго- ритм, про- грамму и раз- работать сис- тему тестов при решении практических задач профес- сиональной деятельности
Выпускник умеет решать ти- повые зада- чи по про- граммирова- нию с ис- пользовани- ем базовых алгоритми- ческих кон- струкций и подпро- грамм;	деятельност- ный – 0,1	Способен понять пра- вильно напи- санные про- граммы по типовым за- дачам	Способен понять и объ- яснить ос- новные мо- менты алго- ритма пра- вильно напи- санных про- грамм для решения ти- повых задач	Способен модифици- ровать алго- ритм пра- вильно на- писанной программы при несуще- ственном изменении условия ти- повой задачи	Способен самостоя- тельно напи- сать алго- ритм и про- грамму при решении типовых за- дач	Способен са- мостоятельно написать алго- ритм, про- грамму и раз- работать сис- тему тестов при решении типовых задач
Выпускник владеет ме- тодами декомпози- ции слож- ных задач на независимые подзадачи	деятельност- ный – 0,2	Не имеет представле- ния о методах декомпози- ции, действу- ет интуитив- но	Имеет слабое представле- ние о деком- позиции, ко- торое не ока- зывает влия- ние на форм- ирование навыков структурного программи- рования	При рас- смотрении правильно работающих программ затрудняется в выделении независимых подзадач	При написа- нии про- грамм может выделить независимую подзадачу и обосновать необходи- мость ее оформления в виде функ- ции пользо- вателя	При програм- мировании всегда выделя- ет независи- мые подзада- чи в функции пользователя, способен к разработке собственной библиотеки функций

Владение знаниями о содержании, основных этапах и тенденций развития программирования, математического обеспечения и информационных технологий (ПК-7)

Планируемые результаты обучения	Критерии оценивания с весовым коэффициентом	Показатели оценивания				
		1	2	3	4	5
Выпускник знает интерфейс среды программирования VS 2010;	когнитивный – 0,1	Знает о существовании среды и возможности ее установки на компьютерах, работающих в OS Windows	Знает о возможности работы в среде в локальном варианте и через удаленный рабочий стол сервера кафедры ИиИТ	Знает алгоритмы создания и построения решений	Знает алгоритмы отладки и тестирования решений	Знает о назначении программ-мастеров среды, уникальных возможностях текстового редактора
Выпускник умеет осуществлять поиск необходимой информации в глобальных компьютерных сетях и LMS MOODLE;	деятельностный – 0,1	Не может осуществлять поиск необходимой информации в глобальных компьютерных сетях и LMS MOODLE	Частично владеет практическими навыками поиска необходимой информации в глобальных компьютерных сетях и LMS MOODLE	Несистематично осуществляет поиск необходимой информации в глобальных компьютерных сетях и LMS MOODLE	Демонстрирует умение использования необходимой информации в глобальных компьютерных сетях и LMS MOODLE	Способен осуществлять поиск необходимой информации в глобальных компьютерных сетях и LMS MOODLE и пополнять знания на основе использования электронных источников.
Выпускник владеет методами отладки и тестирования программ	деятельностный – 0,1	Не имеет представления о назначении отладки и необходимости тестирования	Имеет слабое представление о назначении отладки и необходимости тестирования, применяет в случае крайней необходимости	Несистематично осуществляет отладку и тестирование программ	Демонстрирует умение отладки программ с использованием механизма пошагового выполнения алгоритма; набор тестов для проверки алгоритма выбирается непродуманно	Систематически осуществляет отладку программ с использованием механизма пошагового выполнения различных режимов отладки функций пользователя и точек останова; грамотно продумывает набор тестов для проверки каждой оконечности алгоритма

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

7.1 Основная литература:

1. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования [Текст] / Эрих Гамма, Ричард Хэлм, Ральф Джонсон, Джон Влиссидес – СПб.: Питер, 2016. - 366 с.

7.2 Дополнительная литература:

1. Фридман, А.Л. Объектно-ориентированное программирование на языке Си++ [Текст] / А. Л. Фридман – М.: Горячая Линия - Телеком, 2012. - 234 с.
2. Фримен, Э. Паттерны проектирования [Текст] / Эрик Фримен, Элизабет Фримен – СПб.: Питер, 2016. - 656 с.

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

1. 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>
2. 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
3. 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
4. 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

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина «Языки программирования» направлена на формирование у студентов знаний, умений и владений в области программирования на одном из языков процедурной парадигмы. В результате изучения данной дисциплины студенты должны научиться решать задачи с помощью ЭВМ, раскладывать сложные задачи на независимые подзадачи, представлять, как можно использовать возможности компьютера для решения прикладных задач. Преподавателю необходимо провести систематизацию и выравнивание знаний студентов в области программирования, поскольку они могут сильно варьироваться вследствие того, что часть студентов обучалась по базовому, а часть – по профильному курсу предмета «Информатика и ИКТ» в среднем звене школы.

Преподавание дисциплины должно включать в себя следующие образовательные технологии:

- 1) Организация лекций с использованием при необходимости мультимедийных технологий;
- 2) Использование в ходе лабораторных работ дидактических материалов в виде: опорных конспектов по теоретической составляющей занятий, файлов с примерами программ и т.п.
- 3) Использование ресурсов LMS MOODLE с целью организации процесса систематизации, приобретения и контроля знаний;

- 4) Организация выполнения индивидуальных заданий с целью усиления приобретения навыков самостоятельного построения моделей, разработки алгоритмов и написания программ;
- 5) Организация презентации индивидуальных заданий с целью формирования навыка владения профессиональной терминологией, грамотного выстраивания цепочки высказываний для доказательства обоснованности построенных моделей предметной области и правильно-сти отражения реальных свойств объектов;
- 6) Формирование у студентов убежденности в необходимости последовательного освоения следующих этапов в образовательной деятельности:
 - a. ознакомься с содержанием и теоретическими основами изучаемой темы;
 - b. рассмотри, обсуди с другом и протестируй задачу, решенную кем-то;
 - c. реши самостоятельно задачу, подобную рассмотренной ранее;
 - d. реши самостоятельно задачу по изучаемой теме.

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

Материально-техническое обеспечение дисциплины:

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.);

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 «ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

1. Учебные аудитории для проведения занятий лекционного типа, оборудованные мультимедийными средствами обучения.

2. Учебные аудитории для проведения лабораторных 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 «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»

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: способностью решать задачи профессиональной деятельности в составе научно-исследовательского и производственного коллектива (ПК-4)

В результате освоения дисциплины студент должен приобрести:

знания основных алгоритмических конструкций, способов представления структур данных, классификаций языков программирования, основных требований к языкам программирования, способов реализации языков, основных алгоритмов решения задач по программированию.

умения •выбирать язык и метод программирования в зависимости от решаемой задачи, разрабатывать алгоритм и программу в соответствии с этапами решения задачи и стандартом языка, анализировать и доказывать правильность программы и эффективность алгоритма; грамотно составлять сопроводительную документацию по результатам решения задач.

навыки квалифицированного применения в профессиональной деятельности современных языков и парадигм программирования, технологией разработки алгоритмических и программных решений в области системного и прикладного программирования.

2. Место дисциплины в структуре ОПОП.

Дисциплина «Языки программирования» относится к блоку базовых дисциплин направления подготовки. Изучение данной дисциплины осуществляется в 4 семестре.

3. Объем дисциплины: 5 зачетных единиц.

4. Образовательный процесс осуществляется на русском языке.

5. Разработчики:

Клепиков Алексей Константинович, к.т.н, ст. преподаватель кафедры информатики и информационных технологий.

**13. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ К РАБОЧЕЙ ПРОГРАММЕ
ДИСЦИПЛИНЫ «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»****13. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ К РАБОЧЕЙ ПРОГРАММЕ
ДИСЦИПЛИНЫ «ОСНОВЫ АЛГОРИТМИЗАЦИИ»**

- 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

«26» августа 2016 г.

Программа составлена в соответствии с требованиями ФГОС ВО.

Разработчики:

Фамилия, имя, отчество	Учёная степень	Учёное звание	Должность	Дата разработки	Подпись
Клепиков Алексей Константинович	кандидат технических наук		ст. преподаватель кафедры информатики и информационных технологий;	24.12.2015	