

Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
471086fad29a3b330e244e728abc3661ab35c9d50210def0e75e03a5b6fdf6436
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 технологический
Профилирующая кафедра теории и методики преподавания информатики



И.И. Тимченко
15 февраля 2018г.

Рабочая программа дисциплины
Б1.В.ДВ.06.01 Решение задач по программированию повышенной сложности

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, 2018

Лист внесения изменений

Сведения об утверждении:

утверждена Ученым советом факультета

(протокол Ученого совета факультета № 6 от 3.03.2016)

на 2016 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)

Изменения по годам:

На 2018 год

утвержден (а) Ученым советом факультета

(протокол Ученого совета факультета № 6 от 15.02.2018)

на 2018 год набора

Одобрен (а) на заседании методической комиссии

протокол методической комиссии факультета № 6 от 07.02.2018)

Одобрен (а) на заседании обеспечивающей кафедры ТиМПИ

протокол № 5 от 19.01.2018) Можаров М.С. (Ф. И.О. зав. кафедрой) /  _____ (подпись)

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной образовательной программы «Педагогическое образование» по программе бакалавриата профиля "Информатика"	4
2. Место дисциплины в структуре ООП бакалавриата	5
3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся	5
3.1. Объем дисциплины (модуля) по видам учебных занятий (в часах)	5
4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий	6
4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)	6
4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)	7
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)	8
6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)	10
6.1. Паспорт фонда оценочных средств по дисциплине (модулю)	10
6.2. Типовые контрольные задания или иные материалы	10
6.2.1. Зачет	10
6.2.2. Наименование оценочного средства* (в соответствии с таблицей 6.1)	12
6.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций	15
7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)	16
а) основная учебная литература:	16
б) дополнительная учебная литература:	17
9. Методические указания для обучающихся по освоению дисциплины (модуля)	17
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)	18
Перечень образовательных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю)	18
Занятия, проводимые в интерактивных формах	18
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	19
12. Иные сведения и (или) материалы	19
12.1. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья	19

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной образовательной программы «Педагогическое образование» по программе бакалавриата профиля "Информатика"

В результате освоения ООП бакалавриата обучающийся должен овладеть следующими результатами обучения по дисциплине:

<i>Коды компетенции</i>	Результаты освоения ООП <i>Содержание компетенций *</i>	Перечень планируемых результатов обучения по дисциплине
ПК-4	способностью использовать возможности образовательной среды для достижения личностных, метапредметных и предметных результатов обучения и обеспечения качества учебно-воспитательного процесса средствами преподаваемого предмета	Знать: сущность личностных, метапредметных и предметных результатов обучения; понятие «качество учебно-воспитательного процесса»; основные характеристики и способы формирования безопасной развивающей образовательной среды; специфику общего образования и особенности организации образовательного пространства в условиях образовательной организации; основные психолого-педагогические подходы к проектированию и организации образовательного пространства; способы для достижения личностных, метапредметных и предметных результатов обучения и обеспечения качества учебно-воспитательного процесса средствами преподаваемого учебного предмета; современные педагогические технологии реализации компетентностного подхода с учетом возрастных и индивидуальных особенностей обучающихся; методы и технологии поликультурного, дифференцированного и развивающего обучения. Уметь: применять современные образовательные технологии, включая информационные, а также цифровые образовательные ресурсы для достижения личностных, метапредметных и предметных результатов обучения; разрабатывать и реализовывать проблемное обучение, осуществлять связь обучения по предмету (курсу, программе) с практикой, обсуждать с обучающимися актуальные события современности; разрабатывать и реализовывать программы развития универсальных учебных действий, образцов и ценностей социального поведения; поддерживать в детском коллективе деловую, дружелюбную атмосферу для обеспечения безопасной развивающей образовательной среды.
СПК-2	способен использовать математический аппарат, методологию программирования и современные компьютерные технологии для реализации аналитических и техно-логических 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
--	--	---

2. Место дисциплины в структуре ООП бакалавриата

Данная дисциплина относится к вариативной части базовой части ОПОП и является дисциплиной по выбору.

Требования к входным знаниям и умениям: необходимо пройти обучение по дисциплинам «Математическая логика», «Теория алгоритмов», «Программирование».

Дисциплина изучается на 4 курсе (ах) в 8 семестре (ах).

3. Объем дисциплины 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. Объем 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**):	40	20
в т. числе:		
Лекции	10	4
Семинары, практические занятия		
Практикумы		
Лабораторные работы	30	16
в т.ч. в активной и интерактивной формах	16	8
Внеаудиторная работа (всего**):		
В том числе, индивидуальная работа обучающихся с преподавателем:		

Объём дисциплины	Всего часов	
	для очной формы обучения	для заочной /очно-заочной формы обучения
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**)	68	84
Вид промежуточной аттестации обучающегося (зачет / экзамен****)	Зачет	Зачет

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

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

для очной формы обучения

№ п/п	Раздел дисциплины	Общая трудоём кость (часах)	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах)			Формы текущего контроля успеваемости
			аудиторные учебные занятия		самостоятельн ая работа обучающихся	
		всего	лекции	семинары, практические занятия		
1. Модульное программирование. Программирование абстрактных типов данных						
1.	Процедуры и функции. Модули.	16	2	4	10	Лабораторная работа
2.	Организация динамических структур данных (абстрактных типов данных): списки, стеки, очереди, деревья.	28	2	6	20	Решение индивидуальных задач
2. Объектно-ориентированное программирование						
3.	Реализация абстракций данных методами объектно-ориентированного	22	2	10	10	Решение индивидуальных задач

№ п/п	Раздел дисциплины	Общая трудоём кость (часах)	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоёмкость (в часах)			Формы текущего контроля успеваемости
			аудиторные учебные занятия		самостоятельн ая работа обучающихся	
		всего	лекции	семинары, практические занятия		
	программирования.					
4.	Объектно-событийное и объектно-ориентированное программирование.	42	4	10	28	Решение индивидуальных задач

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

№ п/п	Наименование раздела дисциплины	Содержание
1. Модульное программирование. Программирование абстрактных типов данных		
<i>Содержание лекционного курса</i>		
1.1	Модульное программирование.	
1.2	Программирование абстрактных типов данных	
<i>Темы лабораторных занятий</i>		
1.1	Процедуры, функции. Разработка и вызов подпрограмм в среде Gambas.	
1.2	Изучение внутренней структуры модулей, создание модуля и подключение его к проекту.	
1.3	Многооконные приложения. Создание модальных и немодальных окон в среде Gambas.	
2. Организация динамических структур данных (абстрактных типов данных): списки, стеки, очереди, деревья.		
<i>Содержание лекционного курса</i>		
2.1	Организация динамических структур данных	
<i>Темы лабораторных занятий</i>		
2.1	Простые списки.	
2.2	Неупорядоченные списки. Связанные списки.	
2.3	Стеки.	
2.4	Очереди.	
2.5	Представления деревьев. Обход дерева.	
2.6	Работа с древовидными иерархическими структурами данных.	
2.7	Типы данных record и variant, работа с сеткой строк TStringGrid в среде Gambas.	
3. Реализация абстракций данных методами объектно-ориентированного программирования.		

№ п/п	Наименование раздела дисциплины	Содержание
<i>Содержание лекционного курса</i>		
3.1	Реализация абстракций данных методами объектно-ориентированного программирования.	
<i>Темы лабораторных занятий</i>		
3.1	Абстрактные типы и структуры данных.	
3.2	Классы, объекты, поля, методы.	
3.3	Конструкторы и деструкторы.	
3.4	Свойства и методы объектов.	
3.5	Раннее связывание и позднее связывание.	
4. Объектно-событийное и объектно-ориентированное программирование.		
<i>Содержание лекционного курса</i>		
	Объектно-событийное программирование.	
	Объектно-ориентированное программирование.	
<i>Темы лабораторных занятий</i>		
4.1	Событие и сообщение. Виды событий. События от мыши и клавиатуры.	
4.2	Программирование управления событиями. Организации главного меню, всплывающего меню и панели инструментов.	
4.3	Обработка исключительных событий. Работа с диалогами в среде Gambas.	
4.4	Основы визуального программирования. Компонент.	
4.5	Иерархия компонентов.	

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

Формы контроля: устный (индивидуальный и фронтальный опрос), письменный (выполнение реферативной работы) и лабораторный контроль (решение индивидуальных задач).

Вопросы для устного контроля по темам дисциплины:

1. Параметры-переменные, параметры-значения, параметры-константы. Примеры.
2. Глобальные и локальные идентификаторы. Видимость объектов (идентификаторов). Примеры.
3. Область действия переменных и других идентификаторов.
4. Функции. Их отличие от процедур. Способ обращения к функции. Примеры.
5. Что такое комбинированный тип данных (запись)? Способы описания (примеры).
6. Способы обращения к компонентам записи.
7. Оператор присоединения и его использование.
8. Записи с вариантами. Примеры.
9. Понятие модуля. Разделы модуля.

10. Понятие указателя. Статические и динамические переменные.
11. Карта памяти. Динамическое распределение памяти. Создание и уничтожение динамических переменных.
12. Фрагментация динамической памяти. Освобождение динамической памяти.
13. Примеры использования динамической памяти.
14. Простые списки и действия с ними.
15. Стек. Процедуры обработки.
16. Очередь. Процедуры обработки.
17. Деревья. Формирование дерева. Добавление элемента в дерево. Способы обхода дерева. Исключение компонента из дерева.
18. Понятие об ООП. Отличие ООП подхода к программированию от структурного.
19. Основные понятия ООП. Примеры.
20. Инкапсуляция. Примеры.
21. Наследование. Примеры.
22. Полиморфизм. Примеры.
23. Инкапсулированные в классах поля, методы, свойства.
24. Разделы класса: `published`, `private`, `protected`, `public`.
25. Понятие о визуальном, событийно-управляемом программировании.

Темы рефератов:

1. Простые списки (коллекции, список переменного размера, класс `SimpleList`);
2. Неупорядоченные списки;
3. Связанные списки (добавление элементов, удаление элементов, уничтожение связанного списка, сигнальные метки, инкапсуляция связанных списков, доступ к ячейкам);
4. Разновидности связанных списков (циклические связанные списки, проблема циклических ссылок, двусвязанные списки, потоки);
5. Другие связанные структуры;
6. Псевдоуказатели;
7. Стеки и очереди;
8. Стеки (множественные стеки);
9. Очереди (циклические очереди, очереди на основе связанных списков, применение коллекций в качестве очередей, очереди с приоритетами, многопоточные очереди);
10. Деревья (представления деревьев, полные узлы, списки потомков, представление нумерацией связей, полные деревья, обход дерева);
11. Упорядоченные деревья (добавление элементов, удаление элементов, обход упорядоченных деревьев);
12. Деревья со ссылками (особенности работы);
13. Q-деревья (изменение количества элементов в узле, использование псевдоуказателей, восьмеричные деревья);
14. Сбалансированность дерева;
15. АВЛ-деревья (добавление узла, удаление узла);

16. Б-деревья (производительность Б-деревьев, вставка элементов, удаление элементов, разновидности Б-деревьев);
17. Увеличение производительности Б-деревьев (балансировка, вопросы, связанные с обращением к диску, база данных на основе Б+дерева);
18. Деревья решений (поиск в деревьях игры: минимаксный поиск, оптимизация поиска);
19. Поиск нестандартных решений (метод ветвей и границ, эвристики);
20. Управляющие и контролируемые объекты;
21. Единственный объект. Порождающий объект.

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.	Процедуры и функции. Модули.	СПК-2, ПК-4	Лабораторная работа. Устный опрос.
2.	Организация динамических структур данных (абстрактных типов данных): списки, стеки, очереди, деревья.	СПК-2, ПК-4	Лабораторная работа. Реферат.
3.	Реализация абстракций данных методами объектно-ориентированного программирования.	СПК-2, ПК-4	Лабораторная работа. Устный опрос.
4.	Объектно-событийное и объектно-ориентированное программирование.	СПК-2, ПК-4	Лабораторная работа. Устный опрос.
5.	Зачет		

6.2. Типовые контрольные задания или иные материалы

6.2.1. Зачет

- 1) типовые вопросы (задания)

Примеры задач по теме «Организация динамических структур данных (абстрактных типов данных): стек, очередь, двоичное дерево поиска».

Списки

Вариант 1

Написать программу для реверса списка. Например: список [1, 2, 3] преобразуется в список [3, 2, 1].

Вариант 2

Написать программу для получения значения n-го элемента списка. Например: в списке [three, one, two] второй элемент равен one.

Вариант 3

Написать программу для удаления из списка всех элементов, равных 0. Например: список [1, 0, 2, 0, 0, 3] преобразуется в список [1, 2, 3].

Вариант 4

Написать программу для циклического сдвига списка вправо на заданное число элементов. Например: список [6, 5, 4, 3, 2, 1], циклически сдвинутый вправо на 2 элемента, преобразуется в список [2, 1, 6, 5, 4, 3].

Вариант 5

Написать программу для удаления из списка 2-ого, 4-ого и т.д. элементов. Например: список [6, 5, 4, 3, 2, 1] преобразуется в список [6, 4, 2].

Вариант 6

Написать программу для замены в списке всех элементы, равные 0, на -1. Например: список [1, 0, 0] преобразуется в список [1, -1, -1].

Вариант 7

Написать программу для перевода списка арабских чисел (от 1 до 10) в список римских чисел. Например: список [1, 2, 3] преобразуется в список ["I", "II", "III"].

Вариант 8

Написать программу для подсчета количества определенных элементов в списке. Например: в списке [1, 2, 1, 3, 1] три единицы.

Вариант 9

Написать программу для подсчета количества элементов списка без какого-либо указываемого элемента. Например: в списке [1, 2, 1, 3, 1] два элемента без учета единиц.

Вариант 10

Написать программу для подсчета количества элементов списка, значения которых лежат в определенном диапазоне. Например: в списке [10, 20, 10, 30, 15] два элемента, значения которых больше 10 и меньше 30.

Деревья

Вариант 1

Написать программу для нахождения среднего арифметического листьев вершин бинарного дерева.

Вариант 2

Написать программу для проверки упорядоченности бинарного дерева.

Вариант 3

Вывести бинарное дерево на экран в виде дерева.

Вариант 4

Написать программу для вычисления глубины бинарного дерева (глубина пустого дерева равна 0, глубина одноузловое дерева равна 1).

Вариант 5

Написать программу для подсчета количества листьев вершин дерева, значения которых лежат в определенном диапазоне.

Вариант 6

Написать программу для преобразования дерева в список.

Вариант 7

Написать программу для нахождения среднего арифметического отрицательных узлов дерева.

Вариант 8

Написать программу для подсчета количества вершин бинарного дерева, значения которых не равны 0.

Вариант 9

Написать программу для нахождения среднего арифметического положительных узлов дерева.

Вариант 10

Написать программу для подсчета количества вершин бинарного дерева, значения которых равны 0.

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

1) типовые задания (вопросы) - образец

Структура лабораторной работы представлена следующим образом:

- устный опрос по теме данного лабораторного занятия;
- выполнение лабораторной работы (индивидуальное, в микрогруппах).

Таким образом, на лабораторной работе занятии студент гарантированно получает оценку за опрос и за выполнение лабораторной работы.

Лабораторная работа «Процедуры, функции. Разработка и вызов подпрограмм в среде Gambas».

Gambas позволяет создавать пользовательские процедуры. Основной отличительной чертой пользовательских процедур является то, что они не связаны ни с каким событием и вызов их пользователь осуществляет по своему усмотрению. В виде пользовательской процедуры можно оформить любую подпрограмму и использовать ее в текущем проекте или сохранить на диске и использовать в других программах. Для создания пользовательской процедуры необходимо перейти к секции Главная, в окне программы ввести служебное слово **Sub** и **имя процедуры** и нажать клавишу Enter. После этого появится новая процедура.

Для вызова процедуры используются два способа: по имени процедуры и с помощью оператора Call. При вызове процедуры **по имени** после имени указываются без скобок передаваемые параметры:

ИмяПроцедуры аргумент1, аргумент2, ...

Передаваемые параметры по типам должны соответствовать типам переменных указанных в описании процедуры. В качестве передаваемых параметров могут использоваться строки символов, имена переменных или функций, возвращающих значения заданного типа.

При вызове процедуры с помощью оператор **Call** передаваемые параметры заключаются в скобки:

Call ИмяПроцедуры (аргумент1, аргумент2,...)

Оператор Call используется, как правило, для вызова внешних процедур. Чтобы отличить вызов функции от вызова процедуры, рекомендуют не использовать оператор Call для вызова процедур (при вызове функций передаваемые параметры также заключаются в скобки).

Задание 1. Разработайте приложение в среде Gambas, содержащую процедуру вычисления площади поверхности параллелепипеда.

Процедура, объявленная в секции главная формы

```
Sub SPOverch(a As Single, b As Single, h As Single, s As Single)
s = 2 * (a * b + (a + b) * h)
End Sub
```

```
Private Sub Form_Click()
' Вызывающая процедура
Dim x As Single, y As Single, z As Single, s As Single
x = Val(InputBox("Длина параллелепипеда"))
y = Val(InputBox("Ширина параллелепипеда "))
z = Val(InputBox("высота параллелепипеда "))
' вызов процедуры
SPOverch x, y, z, s
Print Str$(s)
End Sub
```

Наряду с процедурами можно создавать и пользовательские функции, синтаксис которых похож на синтаксис пользовательских.

Синтаксис функции пользователя:

```
Function ИмяФункции ( аргумент1 As тип, [, аргумент2 As тип ] ) As
тип
< операторы >
ИмяФункции = результат
End Function.
```

В качестве аргументов могут быть константы, переменные или выражения. В конце процедуры записывается оператор **ИмяФункции = результат**, здесь **ИмяФункции** – простая переменная, имя которой совпадает с именем функции. Этим оператором имени функции передается вычисленное значение.

Функции пользователя используются так же, как и встроенные функции Visual Basic. То есть, они используются в выражениях справа от знака равно, могут включаться также в оператор Print:

```
< имя переменной > = ИмяФункции ( аргумент 1 [, аргумент 2,
...])
Print ИмяФункции
```

Допускается также использование функции в следующем формате:

```
ИмяФункции (аргумент 1 [, аргумент 2, ...])
```

Внутри функции можно объявлять локальные переменные, указывая их тип: статические или динамические. Переменные уровня формы доступны всем функциям, подключенным к данному модулю или форме. Преждевременный выход из функции осуществляется с помощью оператора **Exit Function**.

Вычислить значение Sin(x) с точностью 0.001. Математическая модель решения данной задачи представляется следующей формулой:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + \frac{x^{2i-1}}{(2i-1)!} + \dots = \sum_{i=1}^{\infty} \frac{x^{2i-1}}{(2i-1)!}$$

Задание 2. Разработайте приложение вычисления факториала с использованием функции. Входным параметром функции будет целое число одинарной длины n – факториал. Функция возвращает целое число двойной длины.

```

Public Function Factorial(n As Integer) As Double
Dim i As Integer, F As Double
F = 1
For i = 1 To n
F = F * i
Next i
Factorial = F
End Function

```

```

Private Sub Form_Click()
Dim n As Integer, F As Double, k As Integer
Dim x As Single, y As Single, s As Single
x = Val(InputBox("Аргумент X"))
e = Val(InputBox("Точность вычисления"))
y = 1 + e: s = 0: k = 1
While Abs(y) >= e
y = (-1) ^ (k - 1) * x ^ (2 * k - 1) / Factorial(2 * k - 1)
s = s + y
k = k + 1
Wend
Print Str$(s)
End Sub

```

Индивидуальные задачи для лабораторного практикума:

Процедуры и функции

1. Напишите программу, состоящую из трех процедур и основной программы. Первая процедура организует ввод двух целых чисел X и Y, вторая проверяет их сумму, третья выводит результат. Используйте эти процедуры в основной программе. Используйте X и Y как глобальные переменные.
2. Напишите программу вычисления площади поверхности и длины экватора на основе известного радиуса планет солнечной системы. Форму планет будем считать шаром. Вычисление площади поверхности и длины экватора оформите отдельными функциями.
3. Напишите программу поиска большего из четырех чисел с использованием подпрограммы поиска большего из двух чисел.
4. Даны координаты вершин многоугольника (x1, y1, x2, y2, ..., x10, y10). Напишите программу для вычисления его периметра (вычисление расстояния между вершинами оформить подпрограммой).
5. Напишите программу вычисления суммы: $1! + 2! + 3! + \dots + n!$, используя функцию вычисления факториала числа k.
6. Напишите программу для вычисления числа сочетаний из N по M. Число сочетаний определяется по формуле $N! / (M! * (N-M)!)$, где N – количество элементов перебора. Используйте подпрограмму вычисления факториала.
7. Напишите программу для определения НОД трех натуральных чисел.
8. Даны действительные числа s, t. Составить программу вычисления выражения $f(t, -2s, 1.17) + f(2.2, t, s-t)$, где $f(a, b, c) = (2a - b - \sin(c)) / (5 + |c|)$.

9. Даны натуральные m и n ($m < n$). Составить программу, сокращающую дробь m/n .
10. Напишите программу вычисления суммы квадратов простых чисел, лежащих в интервале (M, N) .
11. Напишите программу подсчета числа четных цифр, используемых в записи N -значного числа M .
12. Составьте программу вычисления суммы трехзначных чисел, в десятичной записи которых нет четных цифр.
13. Составьте программу вывода на экран всех натуральных чисел, не превосходящих N и делящихся на каждую из своих цифр.
14. Составьте программу нахождения наименьшего натурального N -значного числа X ($X \geq 10$), равного утроенному произведению своих цифр.
15. Составьте программу подсчета числа всех натуральных чисел, меньших M , квадрат суммы цифр которых равен X .

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

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

Максимальное количество баллов, которое может заработать студент за время обучения, равно 100 баллов.

Это предполагает следующие виды заданий:

- 1) Посещение лекционных занятий оценивается по 1 баллу – максимальное количество баллов = 5;
- 2) За правильный ответ по теоретической части лабораторной испытуемый получает 1 балл.

Практическая часть лабораторной работы – 1 балл за каждую задачу, максимальное количество баллов на одной лабораторной работе - 4 балла. За все время обучения 80 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.

- 3) Студент может заработать баллы за дополнительные виды работ:

Написание и защита реферата -4 балла.

Призеры олимпиады по информатике различных уровней -11 баллов.

Для получения зачета по итогам текущей успеваемости студент должен набрать не менее 51 балла.

Распределение баллов по темам и видам учебной работы за семестр

№ п/п	Темы и виды учебной работы	min	max
1	Посещение лекционных занятий	1	5
2	Защита лабораторной работы «Процедуры, функции. Разработка и	2	4

	вызов подпрограмм в среде Gambas»		
3	Защита лабораторной работы «Изучение внутренней структуры модулей, создание модуля и подключение его к проекту»	2	4
4	Защита лабораторной работы «Многооконные приложения. Создание модальных и немодальных окон в среде Gambas»	2	4
5	Защита лабораторной работы «Простые списки»	2	4
6	Защита лабораторной работы «Неупорядоченные списки. Связанные списки»	2	4
7	Защита лабораторной работы «Стеки»	2	4
8	Защита лабораторной работы «Очереди»	2	4
9	Защита лабораторной работы «Представления деревьев. Обход дерева»	2	4
10	Защита лабораторной работы «Работа с древовидными иерархическими структурами данных»	2	4
11	Защита лабораторной работы «Типы данных record и variant, работа с сеткой строк TStringGrid в среде Gambas»	2	4
12	Защита лабораторной работы «Абстрактные типы и структуры данных»	2	4
13	Защита лабораторной работы «Классы, объекты, поля, методы»	2	4
14	Защита лабораторной работы «Конструкторы и деструкторы»	2	4
15	Защита лабораторной работы «Свойства и методы объектов»	2	4
16	Защита лабораторной работы «Раннее связывание и позднее связывание»	2	4
17	Защита лабораторной работы «Событие и сообщение. Виды событий. События от мыши и клавиатуры»	2	4
18	Защита лабораторной работы «Программирование управления событиями. Организации главного меню, всплывающего меню и панели инструментов»	2	4
19	Защита лабораторной работы «Обработка исключительных событий. Работа с диалогами в среде Gambas»	2	4
20	Защита лабораторной работы «Основы визуального программирования. Компонент»	2	4
21	Защита лабораторной работы «Иерархия компонентов»	2	4
22	Защита рефератов(дополн.)		4
23	Участие в олимпиаде (дополн.)		11
	ИТОГО	45	85+15

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

а) основная учебная литература:

- 1) Медведик, В.И. Практика программирования на языке Паскаль (задачи и решения) [Электронный ресурс] : учебное пособие. — Электрон. дан. — Москва : ДМК Пресс, 2013. — 590 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=58700
- 2) Москвитин, А. А. Решение задач на компьютерах. В 2 ч. Ч. I. Постановка (спецификация) задач [Электронный ресурс] : учебное пособие / А. А. Москвитин. - Электрон. текстов.

данные. - Москва ; Берлин : Директ-Медиа, 2015. - 165 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=273666>

3) Москвитин, А. А. Решение задач на компьютерах. В 2 ч. Ч. II. Разработка программных средств. [Электронный ресурс] : учебное пособие / А. А. Москвитин. – Электрон. текстов. данные. - Москва ; Берлин : Директ-Медиа, 2015. - 427 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=273667>

б) дополнительная учебная литература:

- 1) Гусева, Е.Н. Информатика [Электронный ресурс] : учебное пособие. — Электрон. дан. — М. : ФЛИНТА, 2011. — 260 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=44787
- 2) Информатика [Текст] : учебник для вузов / под редакцией В. В. Трофимова ; Санкт-Петербургский государственный университет экономики и финансов (СПбГУЭФ). - Москва : Юрайт, 2011. - 910, [2] с.- (Основы наук). - ISBN 9785991610223
- 3) Новожилов, О. П. Информатика [Текст] : учебное пособие для вузов / О. П. Новожилов. - Москва : Юрайт, 2011. - 564, [12] с. - (Основы наук). - Библиогр.: с. 562-564 (47 назв.). - ISBN 9785991609722

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>
 Газзаев Е. Опыт в программировании на Gambas // <http://ibone.org.ua/gambas/gambas-file-management-1.htm>
 Черный В. Программирование в Gambas // <http://freeschool.altlinux.ru/?p=4234>

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

Вид учебных занятий	Организация деятельности студента
Лекция	Лекции построены на основе использования активных форм обучения: - лекция-беседа (преимущество лекции-беседы состоит в том, что она позволяет привлекать внимание студентов к наиболее важным вопросам темы, определять содержание и темп изложения учебного материала с учетом особенностей студентов), - проблемная лекция (с помощью проблемной лекции обеспечивается достижение трех основных дидактических целей: усвоение студентами теоретических знаний; развитие теоретического мышления; формирование познавательного интереса к содержанию учебного предмета и профессиональной мотивации будущего специалиста), --лекция с заранее запланированными ошибками (Эта форма проведения лекции необходима для развития у студентов умений оперативно анализировать профессиональные ситуации, выступать в роли экспертов, оппонентов, рецензентов, вычленять неверную или неточную информацию). На каждой лекции применяется сочетание этих форм обучения в зависимости от подготовленности студентов и вопросов, вынесенных на лекцию. Присутствие на лекции не должно сводиться лишь к автоматической записи изложения предмета преподавателем. Более того, современный насыщенный материал каждой темы не может (по времени) совпадать с записью в тетради из-за разной скорости процессов – мышления и автоматической записи. Каждый студент должен разработать для себя систему ускоренного фиксирования на бумаге материала лекции.

	Поэтому, лектором рекомендуется формализация записи посредством использования общепринятых логико-математических символов, сокращений, алгебраических (формулы) и геометрических (графики), системных (схемы, таблицы) фиксаций изучаемого материала. Овладение такой методикой, позволяет каждому студенту не только ускорить процесс изучения, но и повысить его качество, поскольку успешное владение указанными приемами требует переработки, осмысления и структуризации материала.
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.

Перечень образовательных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю)

Лекция (информационная, дискуссия, проблемная); лабораторная работа; опрос; работа со справочной системой программ; работа с информационными ресурсами; самостоятельная работа.

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.	Процедуры и функции. Модули.	2		2	Проблемная лекция Работа в малых группах
2.	Организация динамических структур данных (абстрактных типов	2		4	Проблемная лекция Работа в малых группах

	данных): списки, стеки, очереди, деревья.				
3.	Реализация абстракций данных методами объектно-ориентированного программирования.			2	Проблемная лекция Работа в малых группах
4.	Объектно-событийное и объектно-ориентированное программирование.			4	Проблемная лекция Работа в малых группах
	ИТОГО по дисциплине:	4		12	

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

Мультимедийная лекционная аудитория (ноутбук, мультимедийный проектор, интерактивная доска, аудиосистема).

Персональные компьютеры, подключенные к сети Интернет.

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

Для обеспечения образования инвалидов и обучающихся с ограниченными возможностями здоровья разрабатывается адаптированная образовательная программа, индивидуальный учебный план с учетом особенностей их психофизического развития и состояния здоровья.

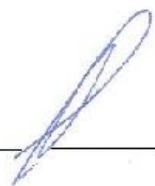
Обучение обучающихся с ограниченными возможностями здоровья осуществляется на основе образовательных программ, адаптированных для обучения указанных обучающихся.

Обучение по образовательной программе инвалидов и обучающихся с ограниченными возможностями здоровья осуществляется факультетом с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

Университетом создаются специальные условия для получения высшего образования по образовательным программам обучающихся с ограниченными возможностями здоровья.

Составитель (и): Можаров М.С. , профессор

(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.