

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра вычислительной техники

“УТВЕРЖДАЮ”
ДЕКАН АВТФ

к.т.н. Рева И. Л.

“ ____ ” _____ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Информатика

Образовательная программа: 09.03.04 Программная инженерия, профиль: Технологии разработки программного обеспечения

Курс: 1, семестр : 1

Факультет автоматизации и вычислительной техники,

		Семестр
№	Вид деятельности	1
1	Всего зачетных единиц (кредитов)	4
2	Всего часов	144
3	Всего занятий в контактной форме, час.	87
4	Лекции, час.	18
5	Практические занятия, час.	0
6	Лабораторные занятия, час.	54
7	из них в активной и интерактивной форме, час.	18
8	Аттестация, час.	2
9	Консультации, час.	13
10	Самостоятельная работа, час.	57
11	Виды самостоятельной работы (курсовой проект, курсовая работа, РГЗ, подготовка к контрольной работе)	РГЗ
12	Вид аттестации	Э

Рабочая программа составлена на основании федерального государственного образовательного стандарта высшего образования по направлению (специальности): 09.03.04 Программная инженерия

ФГОС введен в действие приказом №229 от 12.03.2015 г. , дата утверждения: 01.04.2015 г.

Место дисциплины в структуре учебного плана: Блок 1, базовая

Рабочая программа разработана на основе компетентностной модели выпускника по направлению (специальности): 09.03.04 Программная инженерия

Рабочая программа обсуждена на заседании кафедры

ВТ, протокол заседания кафедры №6 от 20.06.2017

Утверждена на совете факультета автоматики и вычислительной техники, протокол № 6 от 21.06.2017

Программу разработал:

доцент, к.т.н. Романов Е. Л.

Заведующий кафедрой:

доцент, к.т.н. Якименко А. А.

Ответственный за образовательную программу:

доцент Романов Е. Л.

1. Внешние требования

Таблица 1.1

Компетенция ФГОС: ОПК.1 владение основными концепциями, принципами, теориями и фактами, связанными с информатикой; в части следующих результатов обучения:	
з10. знать основы интернет-технологий	
з13. знать основные концепции, принципы, связанные с информатикой	
Компетенция ФГОС: ОПК.2 владение архитектурой электронных вычислительных машин и систем; в части следующих результатов обучения:	
з5. знать основы построения и архитектуры ЭВМ	
Компетенция ФГОС: ОПК.4 способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий; в части следующих результатов обучения:	
з3. знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	
з6. знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	
у11. владеть персональным компьютером как средством управления информацией	
у13. уметь использовать специализированные программные средства при решении профессиональных задач	
у15. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	
у17. уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе	
у18. уметь использовать языки и системы программирования для решения профессиональных задач	
у3. уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	
у6. уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств	
у9. уметь осуществлять поиск информации в локальных и глобальных сетях	
Компетенция ФГОС: ПК.1 готовность применять основные методы и инструменты разработки программного обеспечения; в части следующих результатов обучения:	
з4. знать языки программирования и среды разработки	
з7. знать современные компиляторы, отладчики и оптимизаторы программного кода	
у7. уметь применять методы и приемы отладки программного кода	
у8. уметь выявлять ошибки в программном коде	
Компетенция ФГОС: ПК.3 владение навыками использования различных технологий разработки программного обеспечения; в части следующих результатов обучения:	
у2. уметь использовать методы и приемы алгоритмизации поставленных задач	

2. Требования НГТУ к результатам освоения дисциплины

Таблица 2.1

Результаты изучения дисциплины по уровням освоения (иметь представление, знать, уметь, владеть)	Формы организации занятий
Информатика	
ОПК.1.з10 знать основы интернет-технологий	
1.знать основы интернет-технологий	Самостоятельная работа
ОПК.1.з13 знать основные концепции, принципы, связанные с информатикой	
2.знать основные концепции, принципы, связанные с информатикой	Лекции; Самостоятельная работа
ПК.1.з4 знать языки программирования и среды разработки	

3.знать языки программирования и среды разработки	Лекции; Самостоятельная работа
ПК.1.37 знать современные компиляторы, отладчики и оптимизаторы программного кода	
4.знать современные компиляторы, отладчики и оптимизаторы программного кода	Лекции; Самостоятельная работа
ПК.1.y7 уметь применять методы и приемы отладки программного кода	
5.уметь применять методы и приемы отладки программного кода	Лабораторные работы; Самостоятельная работа
ПК.1.y8 уметь выявлять ошибки в программном коде	
6.уметь выявлять ошибки в программном коде	Лабораторные работы; Самостоятельная работа
ОПК.2.35 знать основы построения и архитектуры ЭВМ	
7.знать основы построения и архитектуры ЭВМ	Лекции; Самостоятельная работа
ПК.3.y2 уметь использовать методы и приемы алгоритмизации поставленных задач	
8.уметь использовать методы и приемы алгоритмизации поставленных задач	Лабораторные работы; Самостоятельная работа
ОПК.4.33 знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	
9.знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	Самостоятельная работа
ОПК.4.36 знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	
10.знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	Самостоятельная работа
ОПК.4.y3 уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	
11.уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	Лабораторные работы
ОПК.4.y6 уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств	
12.уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств	Самостоятельная работа
ОПК.4.y9 уметь осуществлять поиск информации в локальных и глобальных сетях	
13.уметь осуществлять поиск информации в локальных и глобальных сетях	Самостоятельная работа
ОПК.4.y11 владеть персональным компьютером как средством управления информацией	
14.владеть персональным компьютером как средством управления информацией	Самостоятельная работа
ОПК.4.y13 уметь использовать специализированные программные средства при решении профессиональных задач	
15.уметь использовать специализированные программные средства при решении профессиональных задач	Лекции; Лабораторные работы
ОПК.4.y15 уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	
16.уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	Лекции; Лабораторные работы
ОПК.4.y17 уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе	

17. уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе	Лабораторные работы
ОПК.4.у18 уметь использовать языки и системы программирования для решения профессиональных задач	
18. уметь использовать языки и системы программирования для решения профессиональных задач	Лекции; Лабораторные работы

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 1				
Дидактическая единица: Основы компьютерной архитектуры и алгоритмизации				
1. Основы компьютерной архитектуры, языков программирования и системы исполнения программ	0	2	2, 7	Структура программы и языка программирования. Программа = алгоритм + данные. Структура системы исполнения кода. Особенности Си/Си++. Основы компьютерной архитектуры. Прямо адресуемая память, сегментация, система команд, исполнение программы, поток команд. Средства сборки программ в ЯП. Модульное программирование: файлы, библиотеки. Трансляция и компоновка программы. Функция, модуль (файл), проект. Объектный модуль (ОМ), точки входа, внешние ссылки. Компоновка программного модуля из ОМ и библиотек. Программные интерфейсы данных и функций. Заголовочные файлы.
Дидактическая единица: Основы языка программирования Си				
2. Основы языка Си	0	6	3, 4	Структура языка. Уровень описания данных: типы данных и переменные. Представление данных. Байт. Машинное слово. Системы счисления. Двоичная и шестнадцатеричная СС. Уровень преобразования данных. Классификация операций в Си. Операция присваивания. Преобразование типов данных в операциях. Операторы

3. Функции и модульная структура программы	0	4	16, 3, 4	Функции. Формальные и фактические параметры. Способы передачи параметров - по значению и по ссылке. Результат функции. Локальные и глобальные (автоматические и внешние) переменные. Функция как основа модульного программирования. Структура данных как система взаимосвязанных переменных и значений. Последовательность. Стек и очередь. Свойства. Представление стека и очереди в массиве. Использование стека при вызове функции.
Дидактическая единица: Основы технологии разработки программ				
5. Жизненный цикл и технология разработки модуля (функции)	0	2	16, 3, 4	Технология программирования. Программирование методом "северо-западного угла". Образная модель. Предварительный сбор фактов, касающихся программы. Принцип замены формулировки алгоритма управляющей конструкцией языка. Структурное программирование - нисходящее, пошаговое, структурное проектирование программы и данных. Последовательность действий, связанных переменными как основная логическая конструкция алгоритма. Примеры проектирования программ. Сортировка выбором. Варианты: с удалением найденных, со сдвигом оставшейся части, с обменом с очередным элементом. Сбор фактов, структурное проектирование
Дидактическая единица: Традиционные области программирования				
4. Программирование 2D-графики	0	2	15, 18	Библиотека GLUT(OpenGL). Основы 2D-анимации: геометрия, статика, динамика, физика. Исходные данные для выполнения РГР

6. Трудоемкость программ. Сортировка и поиск.	0	2	16, 18	Сортировка и поиск. Понятие записи и ключа. Линейный и двоичный поиск. Трудоемкость алгоритмов сортировки и поиска. Классификация сортировок: выбор, вставка, обмен, подсчет, разделение, слияние. Примеры проектирования программ: разделение, циклическое слияние
--	---	---	--------	--

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 1				
Дидактическая единица: Основы языка программирования Си				
1. Изучение IDE MS Visual Studio	0	4	16, 17	Изучение IDE (NetBeans, VStudio). Создание пустого проекта для консольного приложения. Структура проекта: исходные тексты, заголовочники, объектные модули, программный модуль. Стандартный ввод-вывод (stdio, cprog 4.5), объектная библиотека iostream. Простая программа: ввод массива по предварительно заданной размерности, "красивый" вывод. Отладчик: пошаговое исполнение, точка останова, слежение за переменными
Дидактическая единица: Основы технологии разработки программ				
2. Анализ программ	4	4	16, 18	Изучение материала по анализу программ Проработка стандартных программных контекстов (паттернов структурного проектирования) - 2.4. Задание (контрольная работа №2): каждой бригаде взять по 3 готовых программы из разделов 4.2,4,4.4,4.6 (закладка "программы"), "оживить" и выполнить логический анализ программного кода (стандартные решения и их взаимодействие).
3. Практика анализа программного кода	0	8	16, 18	Оживить фрагменты 1.3-5.8 из "вопросов и ответов" к 2.5 (или л.р.2-3), определить содержательный результат, закомментировать код, отметить все стандартные решения (Сдается одним файлом, защита - объяснение случайно выбранных номеров)

4. Проектирование интерфейса функции (модуля).	4	4	16, 18	Изучить материал по проектированию программ (глава 3). По выданному варианту разработать алгоритм и реализацию программы с использованием технологии структурного программирования.
5. Функции	0	4	16, 18	Из заданий к предыдущей лаб. работе каждой бригаде выдается случайным образом 10 номеров - оформить в виде 10 функций (определить необходимый интерфейс формальных параметров и результата) одним файлом, в main написать по 3 вызова каждой
Дидактическая единица: Традиционные области программирования				
6. Арифметические задачи	0	4	11, 16, 18, 5, 6, 8	Разработка программы решения арифметической задачи (по вариантам). отчет, защита - по 2 номера из "вопросы без ответов" каждому - вызовы, анализ
7. Итерационные циклы	0	4	11, 16, 18, 5, 6, 8	Разработка программы вычисления функции через степенной ряд (по вариантам) - отчет, защита - по 2 номера из "вопросы без ответов" каждому - вызовы, анализ. Для оформления ответа использовать средства обработки данных в Excel, полученные при изучении поведения программы
8. Символы, строки, текст	0	4	11, 16, 18, 5, 6, 8	разработка программы, обрабатывающей строку текста (по вариантам) отчет, защита - по 2 номера из "вопросы без ответов" каждому - вызовы, анализ
9. Сортировка и поиск	0	4	11, 16, 18, 5, 6, 8	разработка программы сортировки по заданному алгоритму (по вариантам), отчет, защита - по 2 номера из "вопросы без ответов" каждому - вызовы, анализ
10. Оценка трудоемкости программы	4	4	11, 15, 16, 18	Трудоемкость программ. Подсчет количества операций, времени выполнения. Перенос данных в Excel. Подбор функции трудоемкости (нормирование) (см. Задания - оценка производительности программы).

11. Программирование графики и 2D-анимация	6	6	15, 16, 18	Программирование графики. Событийное программирование. Моделирование физических явлений (координата, скорость, ускорение, физическая модель, силы, моделирование движения). Прямоугольная и полярная система координат.
12. Текстовые файлы	0	4	16	Работа с текстовыми файлами и стандартными потоками ввода/вывода

Таблица 3.3

Темы для самостоятельного изучения	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 1				
Дидактическая единица: Основы компьютерной архитектуры и алгоритмизации				
1. Основы интернет-технологий	0	10	1, 10, 9	Архитектура, протоколы, сервисы Интернет, основы сетевой безопасности и источники и принципы действия угроз компьютерной безопасности.
2. Основы информационного поиска	0	4	12, 13, 14	Библиографический поиск и поиск в интернете
3. Информация, данные и их форматы	0	10	12, 14	Информация. Определения. Меры. Представление данных в компьютере. Бит (разряд), байт, машинное слово. Представление целых со знаком и без знака, вещественных чисел и символом. Представление мультимедийных данных - аудио, видео, изображение. Растровая форма, сжатие без потери и с потерей качества. Битрейд. Избыточность, архивирование.

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 1				
1	РГЗ	3, 4, 5, 6, 7, 8	17	5
Разработка программы 2D-анимации. Описание геометрии, статики, динамики и физики анимируемого процесса.: Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: http://ermak.cs.nstu.ru/cprog . - Загл. с экрана.				
2	Подготовка к занятиям	3, 4, 5, 6, 8	5	4

Оформление отчетов по лаб. работам, подготовка к лаб. работам: Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: http://ermak.cs.nstu.ru/cprog . - Загл. с экрана.				
3	Подготовка к аттестации	1, 2, 3, 4, 7, 8	11	4
: Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: http://ermak.cs.nstu.ru/cprog . - Загл. с экрана.				
4	Самостоятельное изучение теоретического материала	1, 10, 12, 13, 14, 9	24	0
Студент изучает темы, приведенные в таблице 3.3 : Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: http://ermak.cs.nstu.ru/cprog . - Загл. с экрана.				

5. Технология обучения

Для организации и контроля самостоятельной работы обучающихся, а также проведения консультаций применяются информационно-коммуникационные технологии (табл. 5.1).

Таблица 5.1

Деятельность	Информационно-коммуникационные технологии
Информирование	Социальные сети:YandexDisk vk.com/cprog_cs; Среда электронного обучения НГТУ:ИДО: Си++ от дилетанта до профессионала; Чат:Skype
Размещение учебных материалов	Социальные сети:YandexDisk vk.com/cprog_cs; Среда электронного обучения НГТУ:ИДО: Си++ от дилетанта до профессионала; Чат:Skype

Таблица 5.2

Активные и интерактивные формы проведения занятий

№	Наименование активных форм
1	Тренинг-семинар
Краткое описание применения: Занятие проводится в форме изложения материала с последующим анализом и исполнением тестовых заданий (программных заготовок)	
Подробная информация об использовании технологии приводится в "Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: http://ermak.cs.nstu.ru/cprog . - Загл. с экрана."	

6. Правила аттестации обучающихся по учебной дисциплине

Для аттестации обучающихся по дисциплине используется балльно-рейтинговая система (БРС), позволяющая выставлять оценки по традиционной шкале и 15-уровневой ECTS. Краткая информация о БРС приведена в табл. 6.1.

Таблица 6.1

Оцениваемые виды деятельности обучающихся	Мин. балл	Максимальный балл
Семестр: 1		
Лабораторная:	24	48
РГЗ:	6	12

Экзамен:	0	40
Контролирующие материалы приводятся в "Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: http://ermak.cs.nstu.ru/cprog . - Загл. с экрана."		

В таблице 6.2 представлено соответствие форм контроля заявляемым требованиям к результатам освоения дисциплины.

Таблица 6.2

Коды компетенций ФГОС	Результаты обучения	Формы контроля		
		Защита ЛР	Защита РГЗ	Экзамен
ОПК.1	з10. знать основы интернет-технологий			+
	з13. знать основные концепции, принципы, связанные с информатикой			+
ОПК.2	з5. знать основы построения и архитектуры ЭВМ			+
ОПК.4	з3. знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты			+
	з6. знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе			+
	у11. владеть персональным компьютером как средством управления информацией			+
	у13. уметь использовать специализированные программные средства при решении профессиональных задач		+	
	у15. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов			+
	у17. уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе	+		
	у18. уметь использовать языки и системы программирования для решения профессиональных задач			+
	у3. уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	+		
	у6. уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств			+
	у9. уметь осуществлять поиск информации в локальных и глобальных сетях			+
ПК.1	з4. знать языки программирования и среды разработки		+	+
	з7. знать современные компиляторы, отладчики и оптимизаторы программного кода		+	+
	у7. уметь применять методы и приемы отладки программного кода	+	+	+
	у8. уметь выявлять ошибки в программном коде	+	+	
ПК.3	у2. уметь использовать методы и приемы алгоритмизации поставленных задач	+	+	

Фонд оценочных средств по дисциплине представлен в приложении № 1 к рабочей программе.

7. Литература

Основная литература

1. Подбельский В. В. Программирование на языке Си : учебное пособие для вузов по направлениям: "Прикладная математика и информатика", "Информатика и вычислительная техника", специальностям "Прикладная математика", "Вычислительные машины, комплексы, системы и сети управления" / В. В. Подбельский, С. С. Фомин. - М., 2007. - 600 с. : ил., табл.
2. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы : [учебное пособие для вузов по направлению "Информатика и вычислительная техника" и по специальностям "Вычислительные машины, комплексы, системы и сети", "Программное обеспечение вычислительной техники и автоматизированных систем"] / В. Олифер, Н. Олифер. - СПб. [и др.], 2012. - 943 с. : ил.
3. Конев Ф. Б. Информатика для инженеров : учебное пособие для вузов / Ф. Б. Конев. - М., 2004. - 271, [1] с. : ил., табл
4. Галеева И. С. Интернет как инструмент библиографического поиска : [учебно-практическое пособие] / И. С. Галеева. - СПб., 2007. - 245, [2] с. : ил.
5. Рабинович Е. В. Информатика для всех [Электронный ресурс] : электронный учебно-методический комплекс / Е. В. Рабинович ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: <http://courses.edu.nstu.ru/index.php?show=155&curs=639>. - Загл. с экрана.
6. Культин Н. Б. Основы программирования Microsoft Visual C++ 2010 / Никита Культин. - СПб., 2010. - 376 с. : ил., табл. + 1 CD-ROM.
7. Романов Е. Л. Си/Си ++. От дилетанта до профессионала [Электронный ресурс] : электронное учебное пособие : для 1-2 курсов направления 230100 "Информатика и вычислительная техника" / Романов Е. Л. - Новосибирск, 2010. - 1 электрон. опт. диск (CD-ROM). - Загл. с этикетки диска. - Рег. свидетельство №18891. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000134024
8. Романов Е. Л. Объектно-ориентированное программирование [Электронный ресурс] : электронный учебно-методический комплекс / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: <http://courses.edu.nstu.ru/index.php?show=155&curs=351>. - Загл. с экрана.
9. Романов Е. Л. Си/Си++. От дилетанта до профессионала [Электронный ресурс] : [электронный учебно-методический комплекс] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2010]. - Режим доступа: <http://ermak.cs.nstu.ru/cprog>. - Загл. с экрана.
10. Шахматов Р. Г. Информатика. Ч. 1 : учебное пособие / Р. Г. Шахматов, О. В. Лауферман ; Новосиб. гос. техн. ун-т. - Новосибирск, 2005. - 71, [2] с. : ил.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000046870

Дополнительная литература

1. Романов Е. Л. Практикум по программированию на С++ : [учебное пособие] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - СПб., 2004. - 426, [1] с. : ил.
2. Романов Е. Л. Язык Си++ в задачах, вопросах и ответах : [учебное пособие] / Е. Л. Романов. - Новосибирск, 2003. - 426, [1] с. : ил.
3. Леонтьев Б. К. Форматы файлов Microsoft Windows XP : справочник 2005 / Б. К. Леонтьев. - М., 2005. - 335, [16] с.. - На обл. в подзаг.: HTML, CGML, PDF, TeX, GIF, TIFF, VRML, PND, JPEG, ZIP, UUE, XXE, MIME, WAVE, AVI, QuickTime, MPEG.
4. Лауферман О. В. Информатика. Ч. 2 : учебное пособие / О. В. Лауферман, Р. Г. Шахматов ; Новосиб. гос. техн. ун-т. - Новосибирск, 2008. - 74, [1] с.. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2008/laufer.rar>
5. Юн С. Г. Технология разработки программ и ПО [Электронный ресурс] : электронный учебно-методический комплекс / С. Г. Юн ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: <http://courses.edu.nstu.ru/index.php?show=155&curs=74>. - Загл. с экрана.

6. Коротиков С. В. Технология программирования и разработки программного обеспечения [Электронный ресурс] : электронный учебно-методический комплекс / С. В. Коротиков ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: <http://courses.edu.nstu.ru/index.php?show=155&curs=1255>. - Загл. с экрана.

Интернет-ресурсы

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znaniy.com" : <http://znaniy.com/>
5. :

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

1. Разработка простых консольных приложений с помощью Microsoft Visual Studio 2010 : методические указания к лабораторным работам для 1 курса ФПМИ (направление 010500 - Прикладная математика и информатика, специальность 010503 - Математическое и программное обеспечение и администрирование информационных систем) дневного отделения / Новосиб. гос. техн. ун-т ; [сост.: Ю. В. Тракимус, А. Г. Задорожный, А. В. Чернышев]. - Новосибирск, 2011. - 41, [3] с. : ил., табл.. - Режим доступа: http://www.library.nstu.ru/fulltext/metodics/2011/11_4009.pdf
2. Родников В. В. Лабораторные работы по курсу "Информатика" [Электронный ресурс] : учебно-методическое пособие / В. В. Родников ; Новосиб. гос. техн. ун-т. - Новосибирск, [2012]. - Режим доступа: http://ciu.nstu.ru/fulltext/unofficial/2012/lib_397_1329632332.doc. - Загл. с экрана.

8.2 Специализированное программное обеспечение

- 1 Visual Studio 2013
- 2 Cygwin
- 3 NetBeans IDE

9. Материально-техническое обеспечение

Компьютерный класс

№	Наименование	Назначение
1	Компьютерный класс (Компьютеры объединены в локальную сеть с выходом в Internet)	Компьютеры объединены в локальную сеть с выходом в Internet

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра вычислительной техники

“УТВЕРЖДАЮ”
ДЕКАН АВТФ
к.т.н., доцент И.Л. Рева
“ ” _____ Г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Информатика

Образовательная программа: 09.03.04 Программная инженерия, профиль: Технологии разработки программного обеспечения

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Информатика приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.1 владение основными концепциями, принципами, теориями и фактами, связанными с информатикой	з10. знать основы интернет-технологий	Основы интернет-технологий		Экзамен, вопрос 25
ОПК.1	з13. знать основные концепции, принципы, связанные с информатикой	Основы компьютерной архитектуры, языков программирования и системы исполнения программ		Экзамен, вопросы 1-3
ОПК.2 владение архитектурой электронных вычислительных машин и систем	з5. знать основы построения и архитектуры ЭВМ	Основы компьютерной архитектуры, языков программирования и системы исполнения программ		Экзамен, вопросы 1-3
ОПК.4 способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	з3. знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	Основы интернет-технологий		Экзамен, вопрос 26
ОПК.4	з6. знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	Основы интернет-технологий		Экзамен, вопросы 25-27
ОПК.4	у3. уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	Арифметические задачи Итерационные циклы Оценка трудоемкости программы Символы, строки, текст Сортировка и поиск	Отчет по лабораторной работе, «Итерационные циклы», раздел «Оформление результатов измерений»	

ОПК.4	уб. уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств	Информация, данные и их форматы Основы информационного поиска		Экзамен, вопросы 5,6,27
ОПК.4	у9. уметь осуществлять поиск информации в локальных и глобальных сетях	Основы информационного поиска		Экзамен, вопросы 25
ОПК.4	у11. владеть персональным компьютером как средством управления информацией	Информация, данные и их форматы Основы информационного поиска		Экзамен, вопросы 5,6,25,27
ОПК.4	у13. уметь использовать специализированные программные средства при решении профессиональных задач	Программирование графики и 2D-анимация Программирование 2D-графики	РГЗ, разделы 3-6	
ОПК.4	у15. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	Анализ программ Арифметические задачи Жизненный цикл и технология разработки модуля (функции) Итерационные циклы Оценка трудоемкости программы Символы, строки, текст Сортировка и поиск Текстовые файлы Трудоемкость программ. Сортировка и поиск. Функции и модульная структура программы		Экзамен, задачи
ОПК.4	у17. уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе	Изучение IDE MS Visual Studio	Отчет по лабораторной работе, все разделы	
ОПК.4	у18. уметь использовать языки и системы программирования для решения профессиональных задач	Анализ программ Программирование 2D-графики Трудоемкость программ. Сортировка и поиск.		Экзамен, задачи

ПК.1/ПТ готовность применять основные методы и инструменты разработки программного обеспечения	34. знать языки программирования и среды разработки	Жизненный цикл и технология разработки модуля (функции) Основы языка Си Функции и модульная структура программы	РГЗ, разделы 3-6	Экзамен, вопросы 5-17, задачи
ПК.1/ПТ	37. знать современные компиляторы, отладчики и оптимизаторы программного кода	Жизненный цикл и технология разработки модуля (функции) Основы языка Си Функции и модульная структура программы	РГЗ, разделы 3-6	Экзамен, задачи
ПК.1/ПТ	у7. уметь применять методы и приемы отладки программного кода	Арифметические задачи Итерационные циклы Символы, строки, текст Сортировка и поиск	Отчет по лабораторной работе, РГЗ, разделы 3-6	Экзамен, задачи
ПК.1/ПТ	у8. уметь выявлять ошибки в программном коде	Арифметические задачи Итерационные циклы Символы, строки, текст Сортировка и поиск	Отчет по лабораторной работе, РГЗ, разделы 3-6	
ПК.3/ПТ владение навыками использования различных технологий разработки программного обеспечения	у2. уметь использовать методы и приемы алгоритмизации поставленных задач	Арифметические задачи Итерационные циклы Символы, строки, текст Сортировка и поиск	Отчет по лабораторной работе, РГЗ, разделы 3-6	

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 1 семестре - в форме экзамена, который направлен на оценку сформированности компетенций ОПК.1, ОПК.2, ОПК.4, ПК.1/ПТ, ПК.3/ПТ.

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 1 семестре обязательным этапом текущей аттестации является расчетно-графическое задание (работа) (РГЗ(Р)). Требования к выполнению РГЗ(Р), состав и правила оценки сформулированы в паспорте РГЗ(Р).

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.1, ОПК.2, ОПК.4, ПК.1/ПТ, ПК.3/ПТ, за которые отвечает дисциплина, на разных уровнях.

Общая характеристика уровней освоения компетенций.

Ниже порогового. Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

Пороговый. Уровень выполнения работ отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

Базовый. Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

Продвинутый. Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Паспорт экзамена

по дисциплине «Информатика», 1 семестр

1. Методика оценки

Экзамен проводится в устной форме по билетам. Билет формируется по следующему правилу: один теоретический вопрос из списка и одна задача из списка. Задача реализуется в виде компьютерной программы, исполняемой на тестовых данных.

Форма экзаменационного билета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет АВТФ

Билет № _____

к экзамену по дисциплине «Информатика»

1. Теоретический вопрос. Выражения и операторы. Роль ";" как ограничителя. Классификации управляющей логики программы - последовательность, условие, цикл, переход. Основные операторы Си: if, while, do-while, for, switch, break, continue, return, goto: классификация, особенности синтаксиса и выполнения

2. Задача. Сформировать массив простых чисел в диапазоне от 2 до заданного. Очередное простое число определяется попыткой деления нацело числа на все уже накопленные простые числа.

Утверждаю: зав. кафедрой _____ должность, ФИО
(подпись)

(дата)

2. Критерии оценки

- Ответ на экзаменационный билет (тест) считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не написал текст программы для задачи, оценка составляет <25% базовой.
- Ответ на экзаменационный билет (тест) засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, но не раскрывает их содержание, не может привести примеров использования конструкций языка, имеется текст программы, но она не работает или дает неправильный результат, оценка составляет 25-50% базовой.
- Ответ на экзаменационный билет (тест) билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, раскрывает их содержание, анализирует примеры использования, но не объясняет отдельные

особенности применения, программа компилируется и выполняется, но студент не может внести в нее изменения по требованию преподавателя, оценка составляет 50-75% базовой..

- Ответ на экзаменационный билет (тест) билет засчитывается на **продвинутом** уровне, если студент выполняет все перечисленные выше требования, оценка составляет 75-100% базовой..

3. Шкала оценки

В общей оценке по дисциплине экзаменационные баллы учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины, т.е. 40 баллов за экзамен.

4. Вопросы к экзамену по дисциплине «Информатика»

Теоретические вопросы

1. Структура программы и языка программирования. Программа = алгоритм + данные. Структура системы исполнения кода. Особенности Си/Си++.
2. Основы компьютерной архитектуры. Прямо адресуемая память, сегментация, система команд, исполнение программы, поток команд.
3. Средства сборки программ в ЯП. Модульное программирование: файлы, библиотеки. Трансляция и компоновка программы. Функция, модуль (файл), проект. Объектный модуль (ОМ), точки входа, внешние ссылки. Компоновка программного модуля из ОМ и библиотек. Программные интерфейсы данных и функций. Заголовочные файлы.
4. Позиционная система счисления. Представление чисел. Прямое и обратное преобразование из 10СС в СС с другим основанием. Арифметика в СС с другим основанием (сложение, умножение). Особенности двоичной СС.
5. Единицы представления информации в компьютере: бит, байт, машинное слово. Двоичная и шестнадцатеричная системы счисления и их использование в компьютере. Оценка информационной емкости двоичного числа. Представление целого без знака в машинном слове.
6. Понятие типа данных и переменной. Определение переменных в Си. Базовые типы данных char, int, long как машинные слова. Особенности типа данных char. Знаковая и беззнаковая формы представления в Си. Представление символов. Представление чисел с плавающей запятой.
7. Массивы: особенности работы, инициализация. Массивы как формальные параметры функций.
8. Выражения и операции (обзор и классификация): арифметические, сравнения, логические, машинно-ориентированные, присваивания, адресные, выделения составляющего типа данных. Особенности выполнения операций на Си (совместимость, приоритеты, направление выполнения, действие и результат). Особенности выполнения арифметических операций и операций присваивания. Операция "запятая". Операции сравнения, логические операции.
9. Преобразование базовых типов данных в выражениях: действия, порядок. Явные и неявные преобразования.
10. Выражения и операторы. Роль ";" как ограничителя. Классификации управляющей логики программы - последовательность, условие, цикл, переход. Основные операторы Си: if, while, do-while, for, switch, break, continue, return, goto: классификация, особенности синтаксиса и выполнения.

11. Функции. Формальные и фактические параметры. Способы передачи параметров – по значению и по ссылке. Результат функции. Локальные и глобальные (автоматические и внешние) переменные. Функция как основа модульного программирования.
12. Структура данных как система взаимосвязанных переменных и значений. Стек. Представление стека в массиве. Свойства. Использование стека при вызове функции.
13. Структура данных как система взаимосвязанных переменных и значений. Последовательность. Стек и очередь. Свойства. Представление очереди в массиве.
14. Циклические программы. Виды циклов. Итерационный цикл. Рекуррентные последовательности и итерационные циклы. Программа вычисления корня функции. Программа вычисления суммы степенного ряда.
15. Работа со строками. Представление строки в Си. Строка и массив символов. Поиск в строке. Посимвольная и пословная обработка. Примеры.
16. Преобразования целого и вещественного числа из внешней формы во внутреннюю и обратно.
17. Работа с текстовыми файлами в stdio. Функции открытия/закрытия файла, посимвольный, построчный и форматированный ввод/вывод. Стандартные потоки. (Примеры).
18. Трудоемкость алгоритмов. Время выполнения и трудоемкость. Оценочный характер трудоемкости. Виды функций трудоемкости, чувствительность к данным. Программные конструкции, определяющие различные виды трудоемкости.
19. Сортировка и поиск. Понятие записи и ключа. Линейный и двоичный поиск. Трудоемкость алгоритмов сортировки и поиска.
20. Классификация сортировок: выбор, вставка, обмен, подсчет, разделение, слияние. Идеи. Трудоемкость. (по тестам к защите л.р. Сортировки)
21. Основы анализа программ. “Исторический” и логический анализ программы. “Смысл” выражений и переменных. Стандартные программные контексты. (Показать на примере)
22. Жизненный цикл разработки программы. Образная модель. Предварительный сбор фактов. Технология программирования как «выстраивание» текста программы. «Историческое», структурное и «грязное» программирование (показать на примерах).
23. Идеи (заповеди) структурного программирования – нисходящее, пошаговое, структурное проектирование программы и данных. Последовательность, ветвление и цикл. Модульное программирование (показать на примерах).
24. Время жизни и область действия переменных и функций. Определения и объявления. Глобальные, локальные переменные и их свойства (показать на примерах).
25. Архитектура, протоколы, сервисы Интернет, основы сетевой безопасности и источники и принципы действия угроз компьютерной безопасности.
26. Библиографический поиск и поиск в интернете. правовые основы информационной безопасности и принципы защиты авторского права на программные продукты.
27. Информация. Определения. Меры. Представление данных в компьютере. Бит (разряд), байт, машинное слово. Представление целых со знаком и без знака, вещественных чисел и символом. Представление мультимедийных данных - аудио, видео, изображение. Растровая форма, сжатие без потери и с потерей качества. Битрейд. Избыточность, архивирование.

Задачи

Задачи оформляются в виде функций с передачей всех входных и выходных данных через формальные параметры и результат (глобальные переменные использовать запрещается). В `main` пишется несколько вызовов функции на статических данных, с выводом результатов, демонстрирующих работоспособность программы.

1. Найти наименьшее общее кратное для всех элементов массива - минимальное число, которое делится на все элементы массива без остатка.
2. Написать функцию проверки, является ли заданное число простым. С ее помощью написать функцию поиска простых чисел в диапазоне 1000-2000, две любые части которого - также простые (например, 1997, 1-997, 19-97, 199-7)
3. Сформировать массив простых чисел в диапазоне от 2 до заданного. Очередное простое число определяется попыткой деления нацело числа на все уже накопленные простые числа.
4. Любая целочисленная денежная сумма $n > 7$ может быть выдана без сдачи “трешками” и “пятерками”. Программа находит эти числа для заданного n .
5. На заданном интервале найти все числа, цифры которых находятся в строго возрастающем порядке (например, 532).
6. Число Армстронга – такое число из k цифр, для которого сумма k -х степеней его цифр равна самому этому числу, например $153 = 1^3 + 5^3 + 3^3$. Найти все трехзначные числа Армстронга.
7. Найти все двухзначные (трехзначные) числа, которые совпадают с последними цифрами своих квадратов, например, $25^2 = 625$, $76^2 = 5676$.
8. Число называется совершенным, если оно равно сумме всех своих делителей, например, $6 = 1 + 2 + 3$, $28 = 1 + 2 + 4 + 7 + 14$. Найти все совершенные числа в заданном интервале.
9. Найти все числа в заданном диапазоне, которые делятся на любую из своих цифр.
10. Найти числа в диапазоне 100-10000, для которых куб суммы цифр равен значению самого числа (например, 512 - $5 + 1 + 2 = 8$).
11. Разложить число на простые множители (например $36 = 3 * 3 * 2 * 2$). Результат – последовательность множителей в массиве, ограниченная 0.
12. Определить в массиве максимальную длину последовательности расположенных подряд возрастающих значений и вернуть индекс ее начала и длину.
13. Подсчитать количество слов в строке
14. Найти в строке слово минимальной длины и вернуть индекс его начала.
15. Возвратить индекс начала фрагмента, симметричного относительно центрального символа (например, "abcba"), и имеющего максимальную длину. Длину фрагмента вернуть по ссылке.

16. Возвратить массив индексов начала фрагмента, симметричного относительно центрального символа (например, "abcba"), и имеющих длину ≥ 5 .
17. "Перевернуть" в строке все слова. (Например: "Жили были дед и баба" - "илиЖ илиб дед и абаб").
18. Преобразовать целое из внутренней формы во внешнюю в шестнадцатеричной СС.
19. Преобразовать число в шестнадцатеричной СС из внешней формы во внутреннюю.
20. Преобразовать дробную часть переменной типа double во внешнюю форму представления (строку символов).
21. Найти в строке два одинаковых фрагмента, не содержащих пробелы и имеющих максимальную длину и вернуть индексы их начала
22. В строке находится символ “точка” и символы-цифры дробной части числа. Преобразовать во внутреннюю форму представления (переменную типа double).
23. В строке, содержащей абзац текста, найти концы предложений, обозначенный символом "точка". В следующих за ними словах первую строчную букву заменить на прописную. Между словами количество пробелов может быть любым.
24. Заменить в строке все восьмеричные целые константы на символы с соответствующими кодами, (например, xxx101yuy102zzz на xxx Ayuy102zzz).
25. Найти слово, начинающееся с самой младшей латинской буквы и вернуть индекс его начала.
26. Удалить из строки комментарии вида `"/* ... */"`. Игнорировать вложенные комментарии.
27. Заменить в строке все большие латинские буквы на соответствующие им шестнадцатеричные коды (например, A на 0x41, в константе использовать 2 цифры для представления байта).
28. Заменить в строке все целые константы из любого количества цифр соответствующим повторением следующего за ними символа (например "abc5xacb15y" - "abcsxxxxxasbuuuuuuuuuuuuuuuuuuuuu ").
29. Найти в строке и удалить из нее последовательность повторяющихся символов максимальной длины (например, "absxxxxxasbuuuuuuuuuuuuuuuz" - "absxxxxxasz").
30. Найти в строке наиболее часто встречающийся символ и заменить его на пробел.
31. Найти все вхождения подстроки в строке. Строка и подстрока заданы в массивах символов. Результат – массив, заполненный индексами начала подстроки в строке, последовательность ограничена -1.
32. Найти в строке самую внутреннюю пару скобок и удалить содержащийся между ними текст

Сортировки реализовать в полном соответствии с требованиями варианта.

33. Сортировка выбором. Выбирается минимальный элемент в массиве и запоминается. Затем удаляется, а все последующие за ним элементы до конца массива сдвигаются на один влево. Сам элемент заносится на освободившуюся последнюю позицию.
34. Сортировка выбором. Выбирается минимальный элемент в массиве и запоминается. Затем удаляется, а все последующие за ним элементы массива сдвигаются на один влево. Сам элемент переносится в новый массив.
35. Сортировка выбором. Выбирается минимальный элемент в массиве и запоминается. Затем на его место записывается «очень большое число», например, максимальное $+1$. Сам элемент переносится в новый массив.
36. Сортировка выбором. Выбирается максимальный элемент из оставшихся и меняется с последним из них (обратить внимание на диапазон неупорядоченной части).
37. Сортировка выбором. Выбирается минимальный элемент из оставшихся и меняется с первым из них (обратить внимание на диапазон неупорядоченной части).
38. Сортировка подсчетом. Выходной массив заполняется значениями “-1”. Затем для каждого элемента определяется его место в выходном массиве путем подсчета количества элементов строго меньших данного. Естественно, что все одинаковые элементы попадают на одну позицию, за которой следует ряд значений “-1”. После чего оставшиеся в выходном массиве позиции со значением “-1” заполняются копией предыдущего значения.
39. Сортировка вставками. Берется очередной элемент и извлекается из массива. Затем от начала массива ищется первый элемент, больший данного. Все элементы, от найденного до очередного сдвигаются на один вправо и на освободившееся место помещается очередной элемент. (Поиск места включения от начала упорядоченной части).
40. Сортировка вставками («всплытием»). Берется предпоследний элемент и меняется со следующим, пока не закончится массив и пока следующий будет меньше текущего. Затем берется предыдущий и т.д..
41. Сортировка вставками. Берется очередной элемент массива. Затем от начала выходного массива ищется первый элемент, больший данного. Все элементы, от найденного до последнего сдвигаются на один вправо и на освободившееся место помещается очередной элемент.
42. Сортировка выбором. Выбирается минимальный элемент в массиве и переносится в новый массив. Затем на его место записывается последний элемент исходного массива.

Паспорт расчетно-графического задания (работы)

по дисциплине «Информатика», 1 семестр

1. Методика оценки

В рамках расчетно-графического задания (работы) по дисциплине студенты должны выполнить разработку программы 2D-анимации с использованием библиотеки GLUT, включая описание геометрии, статики, динамики и физики анимируемого процесса.

При выполнении расчетно-графического задания (работы) студенты должны выполнить и оформить в пояснительной записке следующие виды работ:

1. Постановка и анализ задачи;
2. Поиск необходимых сведений из геометрии, статики, физики предметной области, в которой выполняется анимация;
3. Разработка структур данных для представления предметной области и алгоритмов;
4. Написание программного кода;
5. Тестирование программного кода;
6. Оформление текста программы.

Содержание пояснительной записки.

- **Задание.**
- **Структурное описание разработки** должно быть выполнено в виде связного структурированного текста и давать разностороннее представление о программе: какие основные решения приняты при разработке, как работает программа, какие данные являются статическими, какие – динамическими и т.д.. Должны быть упомянуты особенности алгоритмов и дано их содержательное описание. Необходимо использовать структурные схемы и рисунки. Особенно это касается того, что не может быть наблюдаемо непосредственно в тексте программы:
 - основные компоненты и связи между ними;
 - форматы входных и выходных данных, форматы внутренних структур данных;
 - содержательное описание алгоритмов работы на уровне компонент, связей и форматов данных.
- **Функциональное описание.** Содержательное описание типов данных, переменных, интерфейсов функций, классов, фрагментов нетривиальных алгоритмов. Для описания рекомендуется использовать смешанное словесно/формальное представление программы с включением этих элементов в связный текст изложения материала.
- **Описание работы программы** на контрольных примерах или внешний вид программы. Выводы. Ограничения (по памяти, по времени), ошибки, особенности проектирования.

- **Приложение:** исходный текст программы с комментариями по существу алгоритма и структур данных.

В пояснительной записке структурное описание разработки должно давать представление о том, какие основные решения приняты при разработке, как работает программа, какие данные являются статическими, какие – динамическими, должны быть упомянуты особенности алгоритмов. Структурное описание – словесное, с привлечением графических иллюстраций (схем, рисунков), **фрагментов** оригинальных алгоритмов и структур данных.

2. Критерии оценки

Согласно положению о балльно-рейтинговой системе НГТУ, базовый балл рейтинга за РГР определен в рабочей программе (12 баллов). Соответственно, критерий оценки определяется в процентах к этому баллу:

- РГР считается **невыполненной**, если отсутствуют необходимые разделы описания программы, либо программа не работает, либо студент не в состоянии объяснить принципов ее работы (алгоритма) – оценка составляет <25% базовой;
- РГР засчитывается на **пороговом** уровне, разработка не полностью соответствует заданию, имеются ошибки при тестировании, пояснительная записка оформлена со значительными структурными, стилистическими и грамматическими ошибками - оценка составляет не более 25-50% базовой.
- РГР засчитывается на **базовом** уровне, если функционал разработки соответствует заданию, отсутствуют ошибки тестирования, пояснительная записка оформлена в целом грамотно - оценка составляет 50-75% базовой базовой
- РГР засчитывается на **продвинутом** уровне, если функционал реализован полностью и эффективно, в пояснительной записке отражены все аспекты структурного, функционального проектирования и тестирования - оценка составляет 75-100% базовой

3. Шкала оценки

В общей оценке по дисциплине баллы за РГЗ(Р) учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Примерный перечень тем РГЗ(Р)

1. Программа рисования графиков функций 2 переменных в виде поверхности, образованной линиями, параллельными осям x и y.
2. По экрану движется вращающийся куб, изображаемый в виде граней. Вариант: управление вращением с клавиатуры.
3. По экрану движется вращающийся квадрат, отскакивая от границ экрана, в момент отскока скорость вращения возрастает, с течением времени – уменьшается.
4. Падающий текст. Из текстового файла читаются символы, которые «сыплются» с правого верхнего угла экрана. Начальная скорость варьируется в некоторых пределах. Символы «отскакивают» от нижнего края экрана (неупругое соударение).
5. «Насекомое и лампочка». Насекомое летит на источник света таким образом, чтобы угол между источником и направлением вектора скорости был постоянным, т.е. по спирали. Ударяясь о поверхность лампочки, переходит в режим свободного падения с сохранением горизонтальной составляющей скорости при ударе (отскок от поверхности). Насекомые появляются со случайными начальными точками и скоростями.

6. «Реальный маятник». В маятнике, закрепленном на жесткой оси, составляющая силы тяжести, вызывающая угловое ускорение равна $F_0 \cdot \sin(\varphi)$, при малых $\varphi \approx \sin(\varphi)$ – маятник становится математическим. Изобразить график колебаний маятника $y(t)$.
7. «Грузик» раскачивается и колеблется на пружине. Имеется начальное отклонение грузика от $R - R_0$ – от исходного размера пружины и угол поворота φ . На грузик действуют сила сжатия/растяжения пружины и сила тяжести (проекция $m \cdot g \cdot \sin(\varphi)$ вызывает угловое ускорение $m \cdot g \cdot \cos(\varphi)$ – складывается с силой сжатия/растяжения. Изобразить траекторию движения.
8. «Шарики в силовом поле». Шарики движутся по поверхности экрана, отталкиваясь от стенок. На каждую пару шариков действует сила притяжения вида $k/R^2 - k/R_0^2$, т.е. начиная с R_0 притяжение сменяется на отталкивание. (Варианты: 2 шарика, произвольное количество шариков).
9. «Кипящий суп». На дне кастрюли возникают пузырьки, которые всплывают, увеличиваясь в размере. Усложнение: интенсивность появления пузырьков зависит от температуры «дна» T (регулируется) и температуры «воды» (интегрирует разницу $T - T_0$).
10. «Рыбки в аквариуме» плавают, пуская воздушные пузыри.
11. «Светофор».
12. «Шагающий человечек».
13. «Снегопад» - случайным образом падают «снежинки», образуя сугробы по всем поверхностям картинки.
14. «Салют». Движущиеся шарики, увеличивающиеся в размере. Шарики «лопаются», создавая множество мелких, разлетающихся в разные стороны и исчезающих.
15. Анимация сортировок с использованием графической библиотеки. Значение отображается в виде шарика – десятки соответствуют диаметру, единицы – цвету.
16. Анимация обработки строки с использованием графической библиотеки. Символы строки должны изображаться «неровно» (положение, размер) и двигаться неравномерно или с «дрожанием».