

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
Языки программирования

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

Курс: 1, семестр : 2

Факультет автоматики и вычислительной техники,

		Семестр
№	Вид деятельности	2
1	Всего зачетных единиц (кредитов)	4
2	Всего часов	144
3	Всего занятий в контактной форме, час.	64
4	Лекции, час.	18
5	Практические занятия, час.	0
6	Лабораторные занятия, час.	36
7	из них в активной и интерактивной форме, час.	18
8	Аттестация, час.	2
9	Консультации, час.	8
10	Самостоятельная работа, час.	80
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, вариативная

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.3 готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов; в части следующих результатов обучения:	
з3. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	
у6. владеть современными техническими и программными средствами взаимодействия с ЭВМ, технологиями разработки алгоритмов и программ, методами отладки и решения задач на ЭВМ в различных режимах	
Компетенция ФГОС: ПК.1 готовность применять основные методы и инструменты разработки программного обеспечения; в части следующих результатов обучения:	
з12. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	
з4. знать языки программирования и среды разработки	
з7. знать современные компиляторы, отладчики и оптимизаторы программного кода	
з9. знать особенности выбранной среды программирования и системы управления базами данных	
у4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	
у7. уметь применять методы и приемы отладки программного кода	
у8. уметь выявлять ошибки в программном коде	
у9. уметь применять выбранные языки программирования для написания программного кода	
Компетенция ФГОС: ПК.12 способность к формализации в своей предметной области с учетом ограничений используемых методов исследования; в части следующих результатов обучения:	
з5. знать методы и приемы формализации задач	
Компетенция ФГОС: ПК.3 владение навыками использования различных технологий разработки программного обеспечения; в части следующих результатов обучения:	
з5. знать методы и приемы алгоритмизации поставленных задач	
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
Языки программирования	
ПК.1.34 знать языки программирования и среды разработки	
1. знать языки программирования и среды разработки	Лекции; Самостоятельная работа
ПК.1.37 знать современные компиляторы, отладчики и оптимизаторы программного кода	
2. знать современные компиляторы, отладчики и оптимизаторы программного кода	Лекции; Самостоятельная работа
ПК.1.39 знать особенности выбранной среды программирования и системы управления базами данных	
3. знать особенности выбранной среды программирования и системы управления базами данных	Лекции; Самостоятельная работа
ПК.1.312 знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	
4. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	Лекции; Самостоятельная работа
ПК.1.у4 уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	

5.уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	Лабораторные работы; Самостоятельная работа
ПК.1.y7 уметь применять методы и приемы отладки программного кода	
6.уметь применять методы и приемы отладки программного кода	Лабораторные работы; Самостоятельная работа
ПК.1.y8 уметь выявлять ошибки в программном коде	
7.уметь выявлять ошибки в программном коде	Лабораторные работы; Самостоятельная работа
ПК.1.y9 уметь применять выбранные языки программирования для написания программного кода	
8.уметь применять выбранные языки программирования для написания программного кода	Лабораторные работы; Самостоятельная работа
ОПК.3.з3 знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	
9.знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	Лекции
ОПК.3.у6 владеть современными техническими и программными средствами взаимодействия с ЭВМ, технологиями разработки алгоритмов и программ, методами отладки и решения задач на ЭВМ в различных режимах	
10.владеть современными техническими и программными средствами взаимодействия с ЭВМ, технологиями разработки алгоритмов и программ, методами отладки и решения задач на ЭВМ в различных режимах	Лабораторные работы
ПК.3.з5 знать методы и приемы алгоритмизации поставленных задач	
11.знать методы и приемы алгоритмизации поставленных задач	Лекции; Самостоятельная работа
ПК.3.у2 уметь использовать методы и приемы алгоритмизации поставленных задач	
12.уметь использовать методы и приемы алгоритмизации поставленных задач	Лабораторные работы
ПК.12.з5 знать методы и приемы формализации задач	
13.знать методы и приемы формализации задач	Лекции; Самостоятельная работа

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 2				
Дидактическая единица: Типы данных. Иерархия алгоритмов и данных				
1. Указатели и адресная арифметика	1	1	1, 11, 4	Указатели. Указатель как элемент архитектуры компьютера. Синтаксис указателя в Си. Указатель - как "степень свободы" программы. Указатель - формальный параметр и результат функции. Ссылка как неявный указатель. Передача формальных параметров и результата по ссылке. Адресная арифметика. Указатели и массивы. Способы работы через указатель с массивом - индексация и перемещение указателя. Указатели char*, работа со строками через указатели

2. Структурированный тип	1	1	1, 11, 4	Структура. Определение структуры как типа данных. Структура и массив. Массивы структурированных переменных. Указатель на структуру. Указатель на структуру - формальный параметр и результат функции. Инициализация структурированных переменных. Проектирование сложных программ - иерархия типов данных и функций
3. Динамические переменные и массивы	1	1	1, 4	Динамическая память. Динамические переменные и массивы. Операторы и функции управления динамической памятью. Строки - как динамические массивы. "Виртуальные массивы" - программная реализация
4. Иерархия типов данных и переменных	2	2	1, 13, 4	Понятие типа данных. Тип данных и переменная. Базовые и производные типы данных. Виды производных типов данных. Операции извлечения составляющего типа данных. Иерархия определений типов данных и вложенности компонент переменных. Контекстный способ определения типа данных в Си. Примеры анализа контекстных определений. Абстрактный тип данных. Спецификация typedef . Иерархия типов данных и иерархия вызовов функций в модульных программах
5. Модульная организация программы в классическом Си	1	1	1, 2, 4	Модульная организация программы. Время жизни и область действия переменных. Классификация. Определение и объявление переменных. Внешние, автоматические и статические переменные. Область действия функций. Внешние и статические функции. Модульное программирование. Библиотеки. Заголовочные файлы, их назначение и содержание. Файл проекта
Дидактическая единица: Введение в структуры данных				

6. Понятие и виды структур данных.	1	1	11, 13	Понятие структуры данных. Алгоритмическая компонента структуры данных. Статические и динамические структуры данных. Структуры данных "в узком смысле". Упорядоченность. Понятие логического номера элемента, ключевого поля. Операции включения, добавления, удаления, поиска по ключу, извлечения по логическому номеру и сортировки структуры данных. Классификация структур данных: массивы, массивы указателей, списки, деревья. Иерархические структуры данных
7. Массивы указателей	1	1	1, 11, 4, 9	Массивы указателей. Способы формирования массивов указателей - статические, динамические, смешанные. Работа с массивами указателей. Примеры функций включения, исключения, сортировки и поиска. Массив указателей на строки
8. Линейные списки	2	2	1, 11, 4, 9	Списки. Определение элемента списка. Фрагменты алгоритмов работы со списками. Текущий элемент, заголовок. Способы формирования списков. Односвязные списки. Операции включения, исключения, добавления в список. Сортировка списка - выбор, вставка. Представление очереди и стека односвязным списком. Двусвязные списки. Включение и исключение элемента. Проблема концов списка. Циклические списки

9. Рекурсия	2	2	1, 11, 4, 9	Рекурсия. Определение рекурсии. Рекурсивные определения, структуры данных, функции. Особенности проектирования рекурсивных функций. Рекурсивная функция как процесс. Смысл формальных и фактических параметров, локальных и глобальных переменных и результата функции как характеристик рекурсивного процесса. Проектирование рекурсивных алгоритмов и метод математической индукции - принцип "локального мышления". Использование рекурсивных алгоритмах в задачах поиска, основанных на переборе вариантов. Примеры программ движения по лабиринту, расстановки фигур, составления цепочек слов
10. Деревья	2	2	1, 11, 13, 4, 9	Деревья как рекурсивные структуры данных. Определение вершины дерева. Рекурсивный характер обхода дерева. Алгоритмы полного обхода дерева, поиска минимального значения, включения на заданную глубину. Нумерация вершин дерева. Способы нумерации. Поиск элемента по логическому номеру. Способы организации данных в дереве для сокращения глубины просмотра.
11. Деревья двоичного поиска	1	1	1, 11, 13, 4, 9	Двоичное дерево. Включение и поиск в двоичном дереве. Связь двоичного дерева с алгоритмов двоичного поиска. Эффективность поиска в двоичном дереве. Сбалансированность. Извлечение из двоичного дерева по логическому номеру

12. Указатель на функцию и динамическое связывание	1	1	1, 11, 2, 3, 4, 9	Указатель на функцию. Синтаксис. Смысл указателя на функцию - как средства параметризации части алгоритма обработки данных. Указатель на функцию - формальный параметр. Функция вычисления определенного интеграла. Итератор. Итераторы foreach, firstthat. Итераторы, основанные на сравнении - сортировка, поиск минимального, двоичный поиск
13. Двоичные и текстовые файлы	2	2	11, 13, 3	Определение файла. Запись как единица хранения данных. Классификация: текстовые и двоичные, последовательные и произвольного доступа. Модель файла как система представлений о файле в среде программирования, ее связь с языком. Модель текстового файла. Представление текста в различных частях операционной системы. Последовательный текстовый файл. Позиционирование в текстовом файле. Модель двоичного файла. Связь представления двоичного файла с организацией работы с памятью в Си. Распределение памяти в двоичном файле. Чтение, запись и добавление в файл отдельных переменных и массивов. Структуры данных в двоичном файле. Файл записей фиксированной длины как массив переменных одного типа. Файл записей переменной длины как последовательность переменных различных типов или структур данных переменной размерности. Параметризованный двоичный файл - пример хранения таблицы произвольного формата Указатель в файле. Связанные записи в файле. Структуры данных с указателями в файле - массивы указателей, деревья, списки

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 2				
Дидактическая единица: Типы данных. Иерархия алгоритмов и данных				
1. Указатели и адресная арифметика	0	4	10, 12, 5, 6, 7, 8	Разработка программы с использованием указателей, адресной арифметики и динамических массивов
2. Структурированный тип	0	8	10, 12, 5, 6, 7, 8	Определить структурированный тип и набор функций для работы с таблицей записей, реализованной в массиве структур (по вариантам). В перечень функций входят: · ввод записи таблицы с клавиатуры; · загрузка и сохранение таблицы в текстовом файле; · просмотр таблицы; · сортировка таблицы в порядке возрастания заданного поля; · поиск в таблице элемента с заданным значением поля или с наиболее близким к нему по значению; · удаление записи; · изменение (редактирование) записи; · вычисление с проверкой и использованием всех pfgbctq по заданному условию и формуле (например, общая сумма на всех счетах).
Дидактическая единица: Введение в структуры данных				
2. Массивы указателей	0	4	10, 12, 5, 6, 7, 8	Разработка, отладка и тестирование программы, использующей массивы указателей (по вариантам)
3. Линейные списки	0	4	10, 12, 5, 6, 7, 8	Разработка, отладка и тестирование программы, использующей линейные списки (по вариантам)
4. Рекурсия	0	4	10, 12, 5, 6, 7, 8	Разработка, отладка и тестирование программы, основанной на рекурсивных алгоритмах и использующей рекурсивные функции (по вариантам)
5. Деревья	0	4	10, 12, 5, 6, 7, 8	Разработка, отладка и тестирование программы, использующей древовидные структуры (по вариантам)

6. Указатель на функцию, итератор	0	4	10, 12, 5, 6, 7, 8	Разработка программы с использованием указателей на функции для работы с неопределенным типом данных (итератор) (по вариантам)
7. Работа с памятью на низком уровне	0	4	10, 12, 5, 6, 7, 8	Разработка программы упаковки данных по формату с использованием преобразований типа указателя (по вариантам)

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 2				
1	РГЗ	5, 6, 7, 8	30	4
Варианты выполнения РГР: 1. Выполнить 2-3 задания по темам из методического пособия, не охваченных лаб. работами (в скобках - вес в баллах) 9.1. Биты. Байты. Машинные слова (6) 9.2. Работа с памятью на низком уровне (6) 9.4. Позиционирование в текстовых файлах (8) 9.5. Структуры данных в двоичном файле (8) 2. Оценка трудоемкости сортировки (20) (образец в Приложении) 3. Задание к курсовой работе из групп 3, 4, 5, 6, 7, 8 старого учебного плана (список в приложении). : 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	1, 2, 3, 4	35	0
Предварительное ознакомление с материалом лекции, вследствие ее обзорного и проблемного характера и ограничением объема лекционных часов: 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, 11, 13, 2, 3, 4	15	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.				

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: Романов Е. Л. Си/Си++. От дилетанта до профессионала; ЭБС: Романов Е. Л. Си/Си++. От дилетанта до профессионала
Консультирование	Чат: Skype

Размещение учебных материалов	Социальные сети:YandexDisk vk.com/cprog_cs; Среда электронного обучения НГТУ:Романов Е. Л. Си/Си++. От дилетанта до профессионала; ЭБС:Романов Е. Л. Си/Си++. От дилетанта до профессионала
-------------------------------	---

Таблица 5.2

Активные и интерактивные формы проведения занятий

№	Наименование активных форм
1	Проблемный метод:Проблемная лекция
Краткое описание применения: Лекция проводится в форме решения проблемы организации данных или решения задачи поиска, в процессе которого происходит разработка структур данных и алгоритмов	

6. Правила аттестации обучающихся по учебной дисциплине

Для аттестации обучающихся по дисциплине используется балльно-рейтинговая система (БРС), позволяющая выставять 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: 2		
<i>Лабораторная:</i>	20	30
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."		
<i>Контрольные работы:</i>	6	12
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."		
<i>РГЗ:</i>	6	18
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."		
<i>Экзамен:</i>	0	40

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 РГЗ	Экзамен
ОПК.3	з3. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения		+
	уб. владеть современными техническими и программными средствами взаимодействия с ЭВМ, технологиями разработки алгоритмов и программ, методами отладки и решения задач на ЭВМ в различных режимах		+
ПК.1	з12. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования		+
	з4. знать языки программирования и среды разработки		+

	з7. знать современные компиляторы, отладчики и оптимизаторы программного кода		+
	з9. знать особенности выбранной среды программирования и системы управления базами данных		+
	у4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	+	+
	у7. уметь применять методы и приемы отладки программного кода	+	+
	у8. уметь выявлять ошибки в программном коде	+	+
	у9. уметь применять выбранные языки программирования для написания программного кода		+
ПК.12	з5. знать методы и приемы формализации задач		+
ПК.3	з5. знать методы и приемы алгоритмизации поставленных задач		+
	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. Романов Е. Л. Практикум по программированию на C++ : [учебное пособие] / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - СПб., 2004. - 426, [1] с. : ил.
3. Культин Н. Б. Основы программирования Microsoft Visual C++ 2010 / Никита Культин. - СПб., 2010. - 376 с. : ил., табл. + 1 CD-ROM.
4. 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
5. 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.
6. 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.

Дополнительная литература

1. Романов Е. Л. Язык Си++ в задачах, вопросах и ответах : [учебное пособие] / Е. Л. Романов. - Новосибирск, 2003. - 426, [1] с. : ил.
2. 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.
3. 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.

4. Информатика [Электронный ресурс] : учебник / Новосиб. гос. техн. ун-т ; [О. К. Альсова и др.]. - Новосибирск, 2012. - 1 электрон. опт. диск (CD-ROM). - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000175426. - Загл. с этикетки диска.

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

8.2 Специализированное программное обеспечение

1 Visual Studio 2013

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

Языки программирования

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 Языки программирования 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.3 готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	з3. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	Деревья Деревья двоичного поиска Линейные списки Массивы указателей Рекурсия Указатель на функцию и динамическое связывание		Экзамен, вопросы 7-16
ОПК.3	уб. владеть современными техническими и программными средствами взаимодействия с ЭВМ, технологиями разработки алгоритмов и программ, методами отладки и решения задач на ЭВМ в различных режимах	Линейные списки Массивы указателей Структурированный тип Указатели и адресная арифметика		Экзамен, задачи
ПК.1/ПТ готовность применять основные методы и инструменты разработки программного обеспечения	з4. знать языки программирования и среды разработки	Деревья Деревья двоичного поиска Динамические переменные и массивы Иерархия типов данных и переменных Линейные списки Массивы указателей Модульная организация программы в классическом Си Рекурсия Структурированный тип Указатели и адресная арифметика Указатель на функцию и динамическое связывание		Экзамен, вопросы 1-6, задачи
ПК.1/ПТ	з7. знать современные компиляторы, отладчики и оптимизаторы программного кода	Модульная организация программы в классическом Си Указатель на функцию и динамическое связывание		Экзамен, вопросы 2,16
ПК.1/ПТ	з9. знать особенности выбранной среды программирования и системы управления базами данных	Двоичные и текстовые файлы Указатель на функцию и динамическое связывание		Экзамен, вопросы 16-21

ПК.1/ПТ	з12. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	Деревья Деревья двоичного поиска Динамические переменные и массивы Иерархия типов данных и переменных Линейные списки Массивы указателей Модульная организация программы в классическом Си Рекурсия Структурированный тип Указатели и адресная арифметика Указатель на функцию и динамическое связывание		Экзамен, вопросы 1-23, задачи
ПК.1/ПТ	у4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	Линейные списки Массивы указателей Структурированный тип Указатели и адресная арифметика	РГЗ	Экзамен, задачи
ПК.1/ПТ	у7. уметь применять методы и приемы отладки программного кода	Линейные списки Массивы указателей Структурированный тип Указатели и адресная арифметика	РГЗ	Экзамен, задачи
ПК.1/ПТ	у8. уметь выявлять ошибки в программном коде	Линейные списки Массивы указателей Структурированный тип Указатели и адресная арифметика	РГЗ	Экзамен, задачи
ПК.1/ПТ	у9. уметь применять выбранные языки программирования для написания программного кода	Линейные списки Массивы указателей Структурированный тип Указатели и адресная арифметика		Экзамен, задачи
ПК.12/НИ способность к формализации в своей предметной области с учетом ограничений используемых методов исследования	з5. знать методы и приемы формализации задач	Двоичные и текстовые файлы Деревья Деревья двоичного поиска Иерархия типов данных и переменных Понятие и виды структур данных.		Экзамен, вопросы 17-23
ПК.3/ПТ владение навыками использования различных технологий разработки программного обеспечения	з5. знать методы и приемы алгоритмизации поставленных задач	Двоичные и текстовые файлы Деревья Деревья двоичного поиска Линейные списки Массивы указателей Понятие и виды структур данных. Рекурсия Структурированный тип Указатели и адресная арифметика Указатель на функцию и динамическое связывание		Экзамен, вопросы 1-23
ПК.3/ПТ	у2. уметь использовать методы и приемы алгоритмизации поставленных задач	Линейные списки Массивы указателей Структурированный тип Указатели и адресная арифметика		Экзамен, задачи

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 2 семестре - в форме экзамена, который направлен на оценку сформированности компетенций ОПК.2, ПК.33.В, ПК.34.В, ПК.36.В, ПК.37.В, ПК.41.В.

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

В 2 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.2, ПК.33.В, ПК.34.В, ПК.36.В, ПК.37.В, ПК.41.В, 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 «Языки программирования», 2 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 «Языки программирования»

1. Вопрос. Рекурсивные поисковые алгоритмы. Сокращение пространства перебора. Жадные алгоритмы. На примере алгоритма раскраски карты.
2. Задача. Элемент односвязного списка содержит массив указателей на строки. Функция создает структуру данных, читает из файла строки и заполняет ее, пока файл не кончится. (В конце последнего МУ записывается NULL-указатель)

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 «Языки программирования»

Экзаменационные вопросы

1. Тип данных и переменная. Базовые и производные типы данных. Иерархия определений типов данных и вложенности компонент переменных. Контекстный способ определения типа данных в Си. Абстрактный тип данных. Спецификация **typedef**.
2. Модульная организация программы. Время жизни и область действия переменных. Классификация. Определение и объявление переменных. Внешние, автоматические и статические переменные. Область действия функций. Внешние и статические функции.
3. Модульное программирование. Статическое связывание. Библиотеки. Заголовочные файлы, их назначение и содержание. Файл проекта в классическом программировании.
4. Указатели. Указатель как элемент архитектуры компьютера. Синтаксис указателя в Си. Указатель и ссылка. Передача формальных параметров и результата по значению и по ссылке.
5. Адресная арифметика. Указатели и массивы. Способы работы через указатель с массивом.
6. Динамическая память. Динамические переменные и массивы. Операторы и функции управления динамической памятью.
7. Массивы указателей. Способы формирования массивов указателей - статические, динамические, смешанные. Работа с массивами указателей.
8. Списки. Определение элемента списка. Способы формирования списков. Односвязные списки. Двусвязные (циклические) списки.
9. Рекурсия. Рекурсивная структура данных и функция. Реализация рекурсивных функций, роль стека. Инвариант рекурсивной функции. Особенности разработки рекурсивных алгоритмов. Смысл локальных и глобальных переменных, формальных и фактических параметров в рекурсивной функции. Способы накопления результата.
10. Рекурсивные алгоритмы сортировки, трудоемкость.

11. Рекурсивные алгоритмы комбинаторного перебора.
12. Рекурсивные поисковые алгоритмы. Сокращение пространства перебора. Жадные алгоритмы. На примере алгоритма раскраски карты.
13. Деревья. Способы представления деревьев. Полный рекурсивный обход дерева (для всех способов представления). Алгоритмы, основанные на полном рекурсивном обходе дерева.
14. Двоичное дерево. Основные характеристики и алгоритмы. Балансировка дерева.
15. Дерево, упорядоченное по вертикали. Пирамидальная сортировка.
16. Указатель на функцию. Его определение в языке и назначение. Указатель на функцию - формальный параметр. Динамическое связывание. Пример - численное интегрирование.
17. Файл. Двоичный и текстовый файл. Запись. Последовательный и произвольный доступ. Текстовый файл. Позиционирование в текстовом файле. Пример - создание "закладок" в файле.
18. Двоичный файл (ДФ). Физическая модель двоичного файла. Шестнадцатеричный дамп. Функции работы с ДФ, чтение, запись, позиционирование.
19. Последовательный двоичный файл. Понятие формата. Пример: сохранение и загрузка дерева в последовательный ДФ.
20. ДФ произвольного доступа. Распределение памяти в двоичном файле. Файлы записей фиксированной и переменной длины.
21. Указатель в ДФ (абсолютный адрес и смещение). Использование указателей в ДФ на примере массива файловых указателей.
22. Структура. Определение структуры как типа данных. Структура и массив. Массивы структурированных переменных. Указатель на структуру. Указатель на структуру - формальный параметр и результат функции. Инициализация структурированных переменных. Проектирование сложных программ - иерархия типов данных и функций
23. Понятие структуры данных. Алгоритмическая компонента структуры данных. Статические и динамические структуры данных. Структуры данных "в узком смысле". Упорядоченность. Понятие логического номера элемента, ключевого поля. Операции включения, добавления, удаления, поиска по ключу, извлечения по логическому номеру и сортировки структуры данных. Классификация структур данных: массивы, массивы указателей, списки, деревья. Иерархические структуры данных

Практический вопрос (задача). Экзаменационные задачи

Задачу реализовать в виде функции, получающей все данные через параметры. Все структуры данных - динамические.

Замечания по выполняемым операциям. Объединение - результат содержит

элементы из двух исходных структур данных (СД), элемент, присутствующий в обеих СД, включается в одном экземпляре. **Пересечение** - результат содержит элементы, одновременно присутствующие в обеих структурах данных. **Разность** - результат содержит элементы из первой СД, которые отсутствуют во второй.

1. Объединение двух динамических массивов указателей (ДМУ).
2. Пересечение двух динамических массивов указателей.
3. Разность двух динамических массивов указателей.
4. Объединение двух односвязных списков.
5. Пересечение двух односвязных списков.
6. Разность двух односвязных списков.
7. Объединение двух двусвязных списков.
8. Пересечение двух двусвязных списков.
9. Разность двух двусвязных списков.
10. Объединение двух циклических списков.
11. Пересечение двух циклических списков.
12. Разность двух циклических списков.
13. Сортировка односвязного списка выбором.
14. Сортировка односвязного списка вставками.
15. Сортировка двусвязного списка выбором.
16. Сортировка двусвязного списка вставками.
17. Сортировка циклического списка выбором.
18. Сортировка циклического списка вставками.
19. Элемент односвязного списка содержит массив указателей на строки. Функция создает структуру данных, читает из файла строки и заполняет ее, пока файл не кончится. (В конце последнего МУ записывается NULL-указатель).
20. Элемент двусвязного списка содержит массив указателей на строки. Функция создает структуру данных, читает из файла строки и заполняет ее, пока файл не кончится. (В конце последнего МУ записывается NULL-указатель).
21. Элемент циклического списка содержит массив указателей на строки. Функция создает структуру данных, читает из файла строки и заполняет ее, пока файл не кончится. (В конце последнего МУ записывается NULL-указатель).

22. Элемент односвязного списка содержит заголовок односвязного списка (двухуровневый список). Функция создает структуру данных, читает из файла строки и заполняет ее, пока файл не кончится. (длина списка нижнего уровня задана параметром).
23. Двухуровневый массив указателей на строки. Функция создает структуру данных, читает из файла строки и заполняет ее, пока файл не кончится. В конце каждого неполного МУ записывается NULL-указатель в качестве ограничителя.
24. Дерево со списком потомков. Вершина дерева содержит указатель на строку. Полный рекурсивный обход дерева с сохранением адресов строк в динамическом массиве указателей.
25. Дерево со списком потомков. Поиск вершины с максимальным количеством прямых потомков (результат - указатель на вершину).
26. Дерево с массивом указателей на потомков. Вершина дерева содержит указатель на строку. Полный рекурсивный обход дерева с сохранением строк в динамическом массиве указателей.
27. Дерево с массивом указателей на потомков. Поиск вершины с максимальным количеством прямых потомков (результат - указатель на вершину).
28. Дерево с массивом указателей на потомков. Поиск ближайшей вершины со свободным местом (в МУ потомков есть свободное место) (результат - указатель на вершину).
29. Дерево с массивом указателей на потомков. Поиск вершины, наиболее удаленной от корня (результат - указатель на вершину).
30. Двоичное дерево в массиве с вычисляемыми адресами потомков. Включение значения в двоичное дерево.

Паспорт расчетно-графического задания (работы)

по дисциплине «Языки программирования», 2 семестр

1. Методика оценки

В рамках расчетно-графического задания (работы) по дисциплине студенты должны выполнять ряд заданий из методического материала по дисциплине из разделов, не охваченных лабораторными работами, либо выполнить работу по измерению и оценке трудоемкости алгоритма сортировки.

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

1. Постановка и анализ задачи;
2. Разработка структур данных для представления предметной области и алгоритмов;
3. Написание программного кода;
4. Тестирование программного кода;
5. Оформление текста программы.

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 50% базовой
- РГР засчитывается на **базовом** уровне, если функционал разработки соответствует заданию, отсутствуют ошибки тестирования, пояснительная записка оформлена в целом грамотно - оценка составляет 50-80% 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 80-100% 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 (12 баллов).

4. Примерный перечень тем РГЗ(Р)

Варианты заданий к РГР (п.3), варианты заданий к лаб. работам и тестовые задания к защите выложены в <http://ermak.cs.nstu.ru/cprog> и имеются в следующих изданиях (нумерация в соответствии с оглавлением):

«Практикум по программированию на С++: Уч. пособие. СПб:БХВ-Петербург; Новосибирск: Изд-во НГТУ, 2004. - 432 с» или в

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.

Варианты выполнения РГР:

1. Выполнить 2-3 задания по темам из методического пособия, не охваченных лаб. работами (в скобках - вес в баллах)

- 9.2. Работа с памятью на низком уровне (6)
- 9.2. Функции с переменным числом параметров (6)
- 9.4. Позиционирование в текстовых файлах (6)
- 9.5. Структуры данных в двоичном файле (6)

2. Оценка трудоемкости сортировки (20) (образец в Приложении)