

**Правительство Российской Федерации**

**Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
“Национальный исследовательский университет  
“Высшая школа экономики”**

Факультет Бизнес-информатики  
Отделение Прикладной математики и информатики

**Программа дисциплины  
“Основы информатики и программирования”**

для направления 010400.62 "Прикладная математика и информатика"  
подготовки бакалавра

Автор программы  
Дегтярев К.Ю., к.т.н., доцент, [kdegtiarev@hse.ru](mailto:kdegtiarev@hse.ru)

Одобрена на заседании кафедры Управления разработкой программного обеспечения  
« \_\_\_\_ » \_\_\_\_\_ 2013 г.

Зав. кафедрой С.М.Авдошин

Рекомендована секцией УМС по бизнес-информатике « \_\_\_\_ » \_\_\_\_\_ 2013 г.

Председатель Ю.В. Таратухина

Москва, 2013

*Настоящая программа не может быть использована другими подразделениями  
университета и другими вузами без разрешения кафедры-разработчика программы.*



## 1 Область применения и нормативные ссылки

Программа учебной дисциплины "Основы информатики и программирования" (1-й год обучения) устанавливает минимальные требования к знаниям и умениям студента, а также определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих данную дисциплину, учебных ассистентов и студентов, изучающих обязательную дисциплину "Основы информатики и программирования" (блок ИТ.00 - Цикл информационных технологий рабочего учебного плана на 2013-2014 учебный год) по направлению 010400.62 "Прикладная математика и информатика" подготовки бакалавра на факультете Бизнес-информатики (отделение *Прикладной математики и информатики*).

Данная программа разработана в соответствии с

- Федеральным государственным образовательным стандартом высшего профессионального образования по направлению подготовки 010400 «Прикладная математика и информатика» (квалификация (степень) «бакалавр»), введенного в действие приказом МОН РФ № 538 от 20.05.2010 г.,
- образовательным стандартом государственного образовательного бюджетного учреждения высшего профессионального образования "Государственный университет – Высшая школа экономики", в отношении которого установлена категория "национальный исследовательский университет" по направлению подготовки 010400.62 «Прикладная математика и информатика» подготовки бакалавра (утвержден УС НИУ ВШЭ, протокол № 15 от 02.07.10 г., <http://www.hse.ru/data/2012/01/30/1264146097/prim.pdf>),
- образовательной программой направления 010400.62 "Прикладная математика и информатика" подготовки бакалавра,
- базовым учебным планом НИУ ВШЭ по направлению подготовки 010400.62 «Прикладная математика и информатика», утвержденным 04.12.12 г.,
- рабочим учебным планом университета (НИУ ВШЭ) по направлению 010400.62 "Прикладная математика и информатика" подготовки бакалавра, утвержденным в июле 2013 года (<http://ami.hse.ru/rup/archive/?fid=24261>).

## 2 Цели освоения дисциплины

Целями освоения учебной дисциплины "Основы информатики и программирования" являются:

- формирование понимания студентами ключевых положений информатики, ее структуры, связи с другими науками, и программирования на языке высокого уровня (в основе дисциплины лежит язык программирования C++) объяснение базовых правил разработки алгоритмов и программ,
- приобретение студентами уверенных навыков практической работы (программирования) на компьютере, формирование основ для последующих курсов, посвященных созданию современных информационных систем и разработке программ,
- ознакомление студентов с парадигмами (стилями) программирования и механизмами их реализации в программах – основной акцент делается на процедурном программировании и формировании "аккуратного" и постепенного перехода к обсуждению начальных положений объектно-ориентированного программирования (ООП) на примере консольных приложений,
- ориентация студентов на использование современных интегрированных инструментальных сред разработки (в частности, практические занятия дисциплины предусматривают работу в среде разработки MS Visual C++ 2008 (2010 / MS Visual Studio 2012))



Примечание: для самостоятельной работы (установки на персональных компьютерах) студенты МОГУТ использовать другие среды разработки C++ программ – например, Code::Blocks ([www.codeblocks.org](http://www.codeblocks.org)), Orwell Dev-C++ (<http://orwelldevcpp.blogpost.ru>), CodeLite (<http://codelite.org>) и др. – см. стр. 30.

- формирование начальных навыков самостоятельной практической (исследовательской) работы, предполагающей всесторонний анализ поставленной задачи, нахождение (объяснение) возможных путей ее решения и последующего обоснованного выбора конкретной реализации выбранного варианта решения в виде программы на языке C++.

### **3 Компетенции обучающегося, формируемые в результате освоения дисциплины**

В результате освоения дисциплины студент должен

- Знать
  - структуру современной информатики, связь информатики с другими науками, особенности представления числовой (целые и вещественные числа) и символьной информации в компьютере, базовые алгоритмические структуры,
  - основные парадигмы и методологии программирования (процедурный и объектно-ориентированный подходы к программированию),
  - синтаксис и семантику языка C++, лежащего в основе рассматриваемого материала учебной дисциплины; механизмы вызова функций, передачи аргументов в функции; "сложные" и практически значимые составляющие языка C++ и его библиотек (операции приведения к типу, заголовочные файлы, файловые потоки, указатели и работа с ними, использование динамической памяти, динамическая обработка строк, ссылки, работа с последовательными контейнерами (класс vector)),
  - этапы проектирования и разработки программ, возможности интегрированных сред разработки (IDE), использующихся при разработке консольных приложений на языке программирования C++,
  - особенности линейных структур данных (стек и очередь) и их реализацию с использованием связанных списков,
  - рассматриваемые в дисциплине функции стандартной библиотеки C++, элементы STL.
- Уметь
  - представлять числовые и символьные данные в двоичном виде и использовать эти знания для объяснения возникающих в процессе выполнения программ ошибок вычислений,
  - разрабатывать прикладные программы (консольные приложения) с помощью инструментальных интегрированных сред (IDE) и получаемых в процессе изучения дисциплины знаний,
  - отлаживать и тестировать создаваемые программы, используя встроенные средства диагностики IDE,
  - самостоятельно находить решения, необходимые для реализации функциональных требований к программе.
- Иметь навыки (приобрести опыт):
  - решения типовых задач с применением современного и активно используемого языка программирования, возможностей инструментальных средств разработки (IDE),
  - самостоятельной разработки (проектирования), кодирования, отладки, тестирования и документирования консольных программ (т.е. программ, выполняющихся в управляющем окне операционной системы) с использованием процедурного и объектно-ориентированного подходов,



- подготовки подробных отчетов о проведенной работе при решении задач повышенной сложности (домашние задания, предлагаемые для выполнения во втором, третьем и четвертом модулях), обоснования предлагаемых подходов к решению поставленных задач.

В результате освоения дисциплины студент осваивает следующие компетенции:

Компетенция	Код по ФГОС/ НИУ	Дескрипторы – основные признаки освоения (показатели достижения результата)	Формы и методы обучения, способствующие формированию и развитию компетенции
Способность демонстрации общенаучных базовых знаний естественных наук, математики и информатики, понимание основных фактов, концепций, принципов теорий, связанных с прикладной математикой и информатикой	ПК-1	1.1. Понимает основные концепции, принципы, теории и факты, связанные с информатикой (в частности, представление информации в компьютере, архитектуру ЭВМ и т.п.) 1.2. Знает материал основных тем дисциплины, соответствующие реализации (конструкции) языка C++ и библиотечные функции для понимания базовых вопросов и проведения анализа представленных вариантов кода (программ) 1.3. Способен написать код на языке программирования в соответствии с заданным алгоритмом (блок-схема или псевдокод)	Лекции и практические занятия, повторение пройденного материала (консультации) и самостоятельная работа студентов (изучение теоретического материала и практическая реализация программ, предлагаемых в качестве дополнительных заданий)
Способность к формализации в своей предметной области с учетом ограничений используемых методов исследования	ПК-2	2.1. Способен работать с абстракциями (абстракциями данных, процедур (функций), классов и пр.) и адекватно понимать их 2.2. Способен понимать исходную постановку задачи, выявлять ее ключевые моменты, анализировать промежуточные данные и результат (с и без использования компьютера) 2.3. Способен эффективно и быстро применять полученные знания (программирование на языке C++) при решении практических задач, включая нахождение ошибок в программах (их фрагментах) и объяснение путей их устранения	Лекции и практические занятия, повторение пройденного материала (консультации) и самостоятельная работа студентов (изучение теоретического материала и практическая реализация программ, предлагаемых в качестве дополнительных заданий)
Способность решать задачи производственной и технологической деятельности, включая разработку алгоритмических и программных решений в области прикладного программирования	ПК-9	3.1. Демонстрирует уверенные навыки чтения, понимания, выделения главной идеи предлагаемого исходного кода, его фрагмента или описания конкретного примера; способен понимать специфику влияния исходных данных на выполняемые программой действий 3.2. Способен модифицировать и/или исправить исходный код с учетом данных объяснений и требований	Лекции и практические занятия, повторение пройденного материала (консультации) и самостоятельная работа студентов (изучение теоретического материала и практическая реализация программ, предлагаемых в качестве дополнительных заданий). Выполнение



Компетенция	Код по ФГОС/ НИУ	Дескрипторы – основные признаки освоения (показатели достижения результата)	Формы и методы обучения, способствующие формированию и развитию компетенции
			домашних заданий; подготовка отчета о выполненной работе (анализ поставленной задачи, предлагаемые варианты решения, особенности реализации и пр.) и программ на языке C++
Способность применять в профессиональной деятельности современные языки программирования	ПК-10	4.1. Способен понимать и использовать на практике особенности структурного и объектно-ориентированного подходов к программированию, возможности библиотеки C++	Лекции и практические занятия, повторение пройденного материала (консультации) и самостоятельная работа студентов (изучение теоретического материала и практическая реализация программ, предлагаемых в качестве дополнительных заданий) . Выполнение домашних заданий; подготовка отчета о выполненной работе (анализ поставленной задачи, предлагаемые варианты решения, особенности реализации и пр.) и программ на языке C++

#### 4 Место дисциплины в структуре образовательной программы

Настоящая дисциплина является базовой и относится к блоку ИТ.00 – Цикл информационных технологий – рабочего учебного плана направления 010400.62 "Прикладная математика и информатика" подготовки бакалавра на 2013-2014 учебный год.

Изучение данной дисциплины базируется на знаниях студентами математики, основ информатики и алгоритмизации в рамках учебной программы средней школы (предполагается, что большинство студентов 1-го курса успешно сдали итоговый Государственный экзамен по дисциплине «Информатика» – знакомство с основами информатики в объеме школьных дисциплин, наполнение которых бывает весьма различным даже в пределах одного региона страны, может рассматриваться как исключительно полезное для освоения изучаемого материала), умения применять математический аппарат при выборе метода решения поставленной задачи.

Основные положения дисциплины должны (могут) быть использованы в дальнейшем при изучении следующих дисциплин:

- "Алгоритмы и структуры данных" (2-й курс направления 010400.62, ИТ-Б),
- "Современные объектно-ориентированные языки программирования" (2-й курс направления 010400.62, Ф),



- выполнение курсовых работ, предусмотренных РУП по направлению 010400.62).

## 5 Тематический план учебной дисциплины

№	Название темы / лекции	Всего часов по дисциплине	Аудиторные часы		Самостоятельная работа
			Лекции	Практические занятия	
<b>Первый модуль (32 часа / 16 + 16)</b>			сентябрь - октябрь 2013 г.		<b>ЗАЧЕТ в конце модуля</b>
1	Предметная область и основные понятия информатики. Информатика и ее связь с другими науками. Структура современной информатики (вводная лекция)	4	2	-	4
2-3	Организация компьютера. Компьютер как программно-управляемый цифровой автомат. Хранение (представление) информации в компьютере (часть 1)	12	4	4	6
4	Системы счисления. Представление информации в компьютере – целые числа (часть 2); арифметические операции	14	2	4	12
5-6	Представление информации в компьютере – вещественные числа (часть 3). Стандарт двоичной арифметики с плавающей точкой (IEEE 754). Представление символьной информации	16	4	4	12
7-8	Примеры (решение задач и рассмотрение программ на С++). Простые переменные. Как мы подходим к решению проблемы?	12	4	4	8
<b>Второй модуль (32 часа / 16 + 16)</b>			ноябрь - декабрь 2013 г.		<b>КР и ДЗ (текущий контроль)</b>
9-10	Основные алгоритмические структуры. Примеры и обсуждение. Подходы к разработке алгоритмов	12	4	-	10
11	Программирование. Что это такое? Чему и как учиться?	10	2	-	8
12-13	Структурное программирование, нисходящее проектирование, переход к ООП (язык С++). Создание программ. Работа с данными. Выражения и преобразования. Стандартные функции	20	4	4	16



14-15	Составные типы языка C++. Массивы/строки, структуры; операторы языка C++ (часть 1). Указатели. Арифметика указателей (часть 1)	24	4	10	14
16	Указатели и арифметика указателей (часть 2), свободное хранилище, динамическое выделение памяти (начало)	14	2	2	16
<b>Третий модуль (40 часов / 20 + 20)</b> январь - март 2014 г. <b>ЗАЧЕТ [КР и ДЗ (тек. контроль)]</b>					
17-18	Динамическое выделение памяти (продолжение). Проектирование программ. Подпрограммы (часть 1)	20	4	6	14
19-20	Подпрограммы (часть 2). Ссылки (ссылочные переменные)	20	4	6	12
21-22	Рекурсия	18	4	4	16
23-24	Дополнительные примеры (лекции 19-22). Препроцессор	14	4	-	12
25-26	Линейные структуры данных: стек и очередь как примеры абстрактных типов данных	18	4	4	12
<b>Четвертый модуль (40 часов / 20 + 20)</b> апрель - июнь 2014 г. <b>ЭКЗ. [КР и ДЗ (тек. контроль)]</b>					
27-28	Абстракция данных. Понятия класса и объекта.	14	4	2	12
29	Принципы объектно-ориентированного программирования (ООП)	12	2	-	10
30-31	Классы и объекты. Методы класса. Указатели на объекты.	32	4	10	20
32-36	Классы и объекты (окончание). Классы и динамическое выделение памяти (окончание). Класс string. Дополнительные примеры. <i>Заключительные замечания по курсу</i>	38	10	8	20
<b>Итого:</b>		<b>378</b>	<b>72</b>	<b>72</b>	<b>234</b>

## Базовый учебник(и) / ридер(ы) – рекомендуемая литература

### Книги по информатике:

- Акулов О.А., Медведев Н.В. [Информатика: базовый курс](#) : учеб. пособие для студентов вузов, бакалавров, магистров, обучающихся по направлению «Информатика и вычислит. техника», 5-е изд., М.: Омега-Л, 2008.
- Андреева Е.В., Босова Л.Л., Фалина И.Н. [Математические основы информатики. Элективный курс](#), М.: БИНОМ. Лаборатория знаний, 2005.





- **Князева М.Д.** Алгоритмика: от алгоритма к программе. Учебное пособие, М.: КУДИЦ-ОБРАЗ, 2006.
- **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009.
- **Давыдов В.Г.** Программирование и основы алгоритмизации, Высшая школа, 2003.
- **Савельев А.Я.** Основы информатики, М.: изд-во МГТУ им. Баумана, 2001.

Книги по программированию на C++:

- **Хабибуллин И.Ш.** [Программирование на языке высокого уровня. C/C++](#), СПб.: БХВ-Петербург, 2006.
- **Франка П.** [C++. Учебный курс](#), ИД Питер, 2012.
- **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007.
- **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008.
- **Подбельский В.В.** [Стандартный C++](#), М.: Финансы и статистика, 2008. ●●●
- **Прата С.** [Язык программирования C++. Лекции и упражнения](#), пер. с англ., 6-е изд., М.: ООО «И.Д.Вильямс», 2012. ●●●
- **Шилдт Г.** [C++: Базовый курс](#), Вильямс, 2008. ●●●
- **Пахомов Б.И.** [C/C++ и MS Visual C++ 2012 для начинающих](#), СПб.: БХВ-Петербург, 2013.
- **Дейтел Х.М., Дейтел П.Дж.** [Как программировать на C++](#), Бином-Пресс, 2009.
- **Хортон А.** [Visual C++ 2010. Полный курс](#), Вильямс, 2011.
- **Тимофеев В.В.** Самоучитель C++ (C++ как он есть), М.: БИНОМ, 2009.

Публикации в Интернет:

- **Казиев В.М.** Введение в информатику (уровень – для начинающих), Интернет-Университет информационных технологий – дистанционное образование, 2006, <http://www.intuit.ru/department/informatics/intinfo/>
- **Фридман А.Л.** Язык программирования C++, Интернет-Университет информационных технологий – дистанционное образование, 2003, <http://www.intuit.ru/department/pl/cpp/>
- **Роганов Е.А.** Практическая информатика (уровень – для начинающих), Интернет-Университет информационных технологий – дистанционное образование, 2006, <http://www.intuit.ru/department/se/pinform/>
- The C++ Resources Network – [www.cplusplus.com](http://www.cplusplus.com), 2000-2013 (*in English*).
- Справка по C и C++ – <http://ru.cppreference.com/w/>, 2013.
- MSDN C++ Language Reference – <http://msdn.microsoft.com/en-us/library/3bstk3k5.aspx>, Microsoft Corp., 2013 (*in English*).

Дополнительные книги (печатные источники), которые могут быть рекомендованы для самостоятельного изучения:

- **Иванова Г.С.** [Технология программирования](#), М.: КноРус, 2011.
- **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007.
- **Дейтел Х.М., Дейтел П.Дж.** [Как программировать на C++](#), Бином-Пресс, 2008.  
(см. также **Deitel H.M., Deitel P.J.** [C++ How to Program](#), 5<sup>th</sup> ed., Pearson Educ. (Prentice Hall), 2005 (6<sup>th</sup> ed., 2008 или 7<sup>th</sup> ed., 2010); в частности, глава 23 ([Introduction to Standard Template Library](#)) находится в открытом доступе. Кроме этого, можно ознакомиться с краткими, но весьма полезными с





практической точки зрения [Dive into Microsoft Visual C++ .NET](#) и [Dive into Microsoft Visual C++ 6](#) учебными буклетами, подготовленными авторами книги) ●●●

- Deitel P.J., Deitel H.M. [C++11 for Programmers](#) (Dev. Series), 2<sup>nd</sup> ed., Prentice Hall, 2013.
- Шилдт Г. С++ для начинающих, Эком, 2007.
- Шилдт Г. [Самоучитель С++](#), ВHV-Петербург, 3-е изд., 2006.
- Липпман С., Лажоие Ж., Му Б. Язык программирования С++. Вводный курс, Вильямс, 2007.
- Павловская Т.А. [С/С++. Программирование на языке высокого уровня: Учебник для вузов](#), 1-е изд., ИД Питер, 2012.
- Павловская Т.А., Щупак Ю.А. [С/С++. Структурное и объектно-ориентированное программирование. Практикум](#), Питер, 2011.
- Stroustrup B. Programming Principles and Practice using C++, Pearson Education, 2009.  
 (в 2011 г. книга разработчика языка С++ вышла в русском переводе в изд-ве «Диалектика-Вильямс»: Страуструп Б. [Программирование. Принципы и практика использования С++](#), ISBN 978-5-8459-1705-8, 1248 с.)
- Тимофеев В.В. [Самоучитель С++](#), Бином-Пресс, 2009.
- Savitch W., Mock K. [Absolute C++](#), Addison-Wesley Publ., 5<sup>th</sup> ed., 2012 ●●●
- Scheinerman E. C++ for Mathematicians. An Introduction for Students and Professionals, Taylor & Francis Group, 2006.
- Штерн В. [С++](#), изд. Лори, 2013.
- Franek F. [Memory as a Programming Concept in C and C++](#), Cambridge University Press, 2004.

**Замечание:** В данном случае, речь идет лишь о рекомендуемых источниках информации; большое разнообразие книг, посвященных языкам программирования (в частности, С++), в печатном и электронном виде открывает возможности для студентов выбрать именно те книги (документы), которые одновременно максимально полно охватывают материал курса (прежде всего, в той его части, которая описывает программирование на языке) и оказываются удобными (привлекательными) для восприятия студентами с точки зрения организации материала источника и стиля его презентации.

## 6 Формы контроля знаний студентов

Тип контроля	Форма контроля	1 год				Параметры
		1	2	3	4	
Текущий (неделя)	Контрольная работа (КР)	-	7-8 недели – по подгруппам	8-9 недели – по подгруппам	8-9 недели – по подгруппам	практическая работа в классе (программирование) – от 120 до 140 мин. (С.)
	Домашнее задание (ДЗ)	-	3-4 неделя	4-5 неделя	3-4 неделя	письменный отчет с подробным анализом задачи и описанием деталей реализации + программа(ы) – см. <b>Д.</b>
	Работа на практич. занятиях (РПЗ)	-	каждое ПЗ – по подгруппам	каждое ПЗ – по подгруппам	каждое ПЗ – по подгруппам	см. пункт <b>А.</b>
Промежуточный	Зачет (ЗЧ)	■		■		компьютерный тест в 1-м модуле – см. <b>В.</b>
Итоговый	Экзамен (ЭК)				■	



- промежуточный контроль: 2 (два) *зачета (ЗЧ)* в конце 1-го и 3-го модулей,
- итоговый контроль: *экзамен (ЭК)* в конце 4-го модуля (последний модуль курса),
- текущий контроль: 3 (три) *контрольных работы (КР)* в течение учебного года,  
3 (три) *домашних задания (ДЗ)* в течение учебного года.

Контрольные работы (*КР* – их основную часть составляет подготовка C++ программ на компьютере) проводятся во втором (*КР2*), третьем (*КР3*) и четвертом (*КР4*) модулях курса. Домашние задания (*ДЗ*) выдаются студентам в каждом модуле дисциплины, начиная со второго (обозначены по номеру модуля как *ДЗ2*, *ДЗ3* и *ДЗ4*, соответственно).

Оценки промежуточного и итогового контроля складываются из следующих элементов:

**А. работа на практических занятиях (РПЗ) во 2-м, 3-м и 4-м модулях**

контролируется посредством оценивания результатов непродолжительных (до получаса) тестовых работ (*on-fly testing*) и/или качества выполнения предлагаемых самостоятельных заданий в течение времени, отведенного под практические занятия (*class work*). Из-за специфики практических занятий первого модуля, ориентированных, в значительной степени, на объяснение материала и знакомство с языком, *оценивание работы студентов в компьютерном классе в первом модуле НЕ производится*. Непосещение студентом 3-х и более практических занятий в течение любого модуля (второй, третий или четвертый) автоматически приводит к получению оценки «0» (ноль) за Работу на Практических Занятиях (РПЗ) в данном конкретном модуле. В результате, до конца учебного года каждый студент получает 3 (три)  $РПЗ_{(мод\ i)}$  оценки по 10-балльной шкале, по одной за каждый из трех модулей ( $i=2,3,4$ ). В частности, работа студентов на практических занятиях может оцениваться символьными оценками **А** (*отлично*), **В** (*хорошо*), **С** (*удовлетворительно*), **Д** (*плохо*), **Ф** (*пропуск занятия*) и их «оттенками» (например, **В-** или **С+**), которые в конце модуля конвертируются в итоговую оценку (число) по 10-балльной шкале,

**В. зачет (ЗЧ) в конце 1-го модуля**

предусматривает проведение компьютерного тестирования (сессия «вопросы-ответы»), которая организуется для всех без исключения студентов курса. Тематика предлагаемых контрольных заданий (вопросов) охватывает темы курса, которые обсуждаются на лекционных и практических занятиях до момента проведения тестирования. Количество и сложность включенных в работу (программу тестирования) вопросов (решение принимается преподавателями курса) определяет ее продолжительность ( $\leq 80$  минут); оценивание, выполняемое автоматически программой, базируется на 10-балльной шкале,

**С. контрольные работы (КР) в течение 2-го, 3-го и 4-го модулей**

предусматривают, как правило, подготовку (написание, отладку и тестирование) программ в компьютерном классе. Контрольная работа проводится на последнем(-их) занятии(-ях) подгруппы в соответствующем модуле. Каждая выполненная работа в электронном виде представляется на проверку преподавателю; работа (*КР2*, *КР3* и *КР4*) студента оценивается по 10-балльной шкале. Контрольная работа *КР3* или *КР4* может, по решению преподавателей, дополнительно предусматривать проведение компьютерного теста *T* до начала собственно работы над программой (*КР*); если студентам предлагается такой тест (решение преподавателей дисциплины), то оценка за соответствующую контрольную работу вычисляется как  $КРi = 0.25 \cdot T + 0.75 \cdot \overline{КРi}$  ( $i = 3, 4$ ). *КР4* рассматривается в качестве экзаменационной работы (*ЭК*).



#### **Д. домашние задания (ДЗ) во 2-м, 3-м и 4-м модулях курса**

выдаются студентам в течение соответствующего модуля; каждое из заданий (*ДЗ2*, *ДЗ3* и *ДЗ4*) предусматривает подробный анализ, разработку алгоритма решения и его программную реализацию применительно к одной-двум проблемам (задачам). По каждому домашнему заданию студент подготавливает отчет в электронном виде; ориентировочное время, выделяемое на выполнение каждого домашнего задания (*ДЗ*) и подготовку соотв. отчета, составляет не более 2-х недель (как правило, от 11-ти до 13-ти календарных дней). Представляемые студентами отчеты проверяются и оцениваются совместно преподавателями и учебными ассистентами; каждое домашнее задание (*ДЗ*) оценивается по 10-балльной шкале. Несоблюдение установленных сроков и правил представления отчетов (включаются в текст задания) автоматически приводит к получению оценки «0» («не зачтено») за соответствующее *ДЗ*.

Окончательные оценки промежуточного контроля (*ЗЧ1* и *ЗЧ3*) и итогового контроля (*ЭК*) по 10-балльной шкале формируются следующим образом:

**(модуль – этап 1)** оценка зачета *ЗЧ1* (*Промежуточная Оценка 1, ПО1*) определяется 10-балльной оценкой, полученной студентом за компьютерное тестирование (см. выше по тексту). Студент получает оценку «зачет», если автоматически выставляемая оценка (*ЗЧ1*)  $\geq 4$  (см. таблицу соответствия ниже по тексту),

**(модуль – этап 3)** оценка зачета *ЗЧ3* (*Промежуточная Оценка 2, ПО2*) с округлением до ближайшего целого (после вычисления результата) за 3-й модуль определяется как взвешенная сумма

$$ЗЧ3(ПО2) = (0.1 \cdot (РПЗ_{(мод.2)} + РПЗ_{(мод.3)}) + 0.2 \cdot (КР2 + КР3) + 0.25 \cdot ДЗ2 + 0.15 \cdot ПО1),$$

где  $(0.1 \cdot (РПЗ_{(мод.2)} + РПЗ_{(мод.3)}) + 0.2 \cdot КР2 + 0.25 \cdot ДЗ2 + 0.15 \cdot ПО1)$  представляет собой накопленную оценку за 1-й, 2-й и 3-й модули (ее вес в **результатирующей оценке за третий модуль** составляет **80%**); *КР3* представляет собой зачетную работу за 3-й модуль (ее вес составляет **20%**). Студент получает оценку «зачет», если оценка (*ЗЧ3*)  $\geq 4$  (см. таблицу соответствия ниже по тексту).

**(модуль – этап 4)** промежуточная оценка за 4-й модуль (*Промежуточная Оценка 4, ПО4*), учитывающая и оценку за *ДЗ* третьего модуля<sup>1</sup> (по 10-балльной системе), определяется следующим образом:  $ПО4 = (0.2 \cdot РПЗ_{(мод.4)} + 0.35 \cdot ДЗ3 + 0.45 \cdot ДЗ4)$ . В результате, накопленная итоговая (по всем модулям дисциплины) оценка **НО[1234]** студента вычисляется как взвешенная сумма оценок *ПО2* (здесь учтена оценка за 1-й модуль) и *ПО4*, полученных на всех предыдущих этапах, т.е.  $НО[1234] = 0.55 \cdot ПО2 + 0.45 \cdot ПО4$ .

■ Результатирующая итоговая оценка **ИТОГ** за дисциплину с округлением (после получения результата) до ближайшего целого числа баллов (10-балльная система) формируется следующим образом:  $ИТОГ = 0.35 \cdot КР4 + 0.65 \cdot НО[1234]$ , где *КР4* представляет собой зачетную (экзаменационную работу) за 4-й модуль (ее вес составляет **35%** в итоговой оценке).

Перевод итоговой оценки (результатирующая оценка за экзамен) в оценку по пятибалльной шкале осуществляется в соответствии со следующей таблицей:

<sup>1</sup> Домашнее задание (*ДЗ3*) выполняется во второй половине 3-го модуля, и проверка представленных работ происходит в третьем и начале четвертого модуля учебного года.



Таблица соответствия оценок по десятибалльной и пятибалльной системам

По десятибалльной шкале	По пятибалльной шкале
1 – неудовлетворительно 2 – очень плохо 3 – плохо	неудовлетворительно – 2
4 – удовлетворительно 5 – весьма удовлетворительно	удовлетворительно – 3
6 – хорошо 7 – очень хорошо	хорошо – 4
8 – почти отлично 9 – отлично 10 – блестяще	отлично – 5

## 7 Содержание программы

### ● Тема (Лекция) 1: Предметная область и основные понятия информатики

#### ◆ Содержание лекции:

- Общие комментарии, касающиеся курса. Что изучает информатика?
- Состав (структура) информатики. Понятие информации как одного из фундаментальных понятий в современной науке. Эволюция информатики как науки. Связь информатики с другими науками (место информатики в системе наук).

#### ◆ Основная литература

- **Акулов О.А., Медведев Н.В.** Информатика: базовый курс : учеб. пособие для студентов вузов, бакалавров, магистров, обучающихся по направлению «Информатика и вычислит. техника», 5-е изд., М.: Омега-Л, 2008 [стр. 8-36].
- **Симонович С.В.** Общая информатика, СПб.: Питер, 2007 [стр. 8-76].

#### ◆ Дополнительная литература

- **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 8-17, 22-25].
- **Казиев В.М.** Введение в информатику. *Раздел (лекция) 1 – Введение. История, предмет, структура информатики*, Интернет-Университет информационных технологий, 2006 (<http://www.intuit.ru/department/informatics/intinfo/1/>).
- **Казиев В.М.** Введение в информатику. *Раздел (лекция) 15 – Информатизация общества, информационное общество. Интернет*, Интернет-Университет информационных технологий, 2006 (<http://www.intuit.ru/department/informatics/intinfo/15/>).

### ● Тема (Лекции) 2-3: Организация компьютера. Компьютер как программно-управляемый цифровой автомат. Представление информации в компьютере (часть 1)

#### ◆ Содержание лекций:

- Идея хранимой в памяти программы (Дж. фон Нейман). Подсистемы вычислительного устройства.



- Уровни вычислительной системы. Информационный уровень. Организация памяти (ячейки, адрес). Биты и их хранение в памяти. Логические (булевы) операции. Представление информации в компьютерах. Обсуждение примеров.
- ◆ Основная литература
  - **Савельев А.Я.** Основы информатики, М.: изд-во МГТУ им. Баумана, 2001 [стр. 30-36].
  - **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 36-54, 485-508].
  - **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002 [стр. 3-36, 119-127].
  - **Давыдов В.Г.** Программирование и основы алгоритмизации, Высшая школа, 2003.
- **Тема (Лекция) 4: Системы счисления. Представление информации в компьютере – целые числа (часть 2); арифметические операции**
- ◆ Содержание лекции:
  - Понятие системы счисления (ССЧ). Позиционные и непозиционные системы счисления. Основание позиционной системы счисления (p-ичная система счисления). Двоичная, восьмеричная и шестнадцатеричная системы счисления.
  - Перевод чисел из одной системы счисления в другую (перевод  $A_p \rightarrow A_d$  из p-ичной системы счисления в d-ичную) – начало. Схема Горнера (сам. изучение).
  - Целые числа без знака. Целые числа со знаком. Прямой и дополнительный коды. Арифметические операции над целыми числами. Формальные правила двоичной арифметики (сам. изучение). Обсуждение примеров.
- ◆ Основная литература
  - **Акулов О.А., Медведев Н.В.** Информатика: базовый курс : учеб. пособие для студентов вузов, бакалавров, магистров, обучающихся по направлению «Информатика и вычислит. техника», 5-е изд., М.: Омега-Л, 2008 [стр. 61-66, 71-80, 88-103].
  - **Савельев А.Я.** Основы информатики, М.: изд-во МГТУ им. Баумана, 2001 [стр. 63-108].
  - **Андреева Е.В., Босова Л.Л., Фалина И.Н.** Математические основы информатики. Элективный курс, М.: БИНОМ. Лаборатория знаний, 2005 [стр. 12-19, 25-28, 38-49, 73-87].
- ◆ Дополнительная литература
  - **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 26-33].
  - **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002.
  - **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 60-76].
  - **Роганов Е.А.** Основы информатики и программирования. Раздел (лекция) 4 – Особенности представления чисел в ЭВМ, Интернет-Университет информационных технологий, 2006 (<http://www.intuit.ru/department/se/oip/4/>).





- **Симонович С.В.** Общая информатика, СПб.: Питер, 2007 [стр. 171-181].
  - **Казиев В.М.** Введение в информатику. *Раздел (лекция) 4 – Системы счисления и действия в них*, Интернет-Университет информационных технологий, 2006 (<http://www.intuit.ru/department/informatics/intinfo/4/>).
- **Тема (Лекции) 5-6: Представление информации в компьютере – вещественные числа (часть 3). Стандарт двоичной арифметики с плавающей точкой (IEEE 754). Представление символьной информации**
- ◆ **Содержание лекций:**
    - Представление вещественных чисел. Нормализованная запись числа. Формат представления вещественных чисел (одинарный, двойной и расширенный). Обсуждение примеров.
    - Представление символьной (текстовой) информации в компьютерах. Информационный объем блока информации. ASCII код. Расширение ASCII-кодировки (кодировочные таблицы КОИ-8, Windows-1251).
    - Стандарт кодирования Unicode (*сам. изучение*).
  - ◆ **Основная литература**
    - **Акулов О.А., Медведев Н.В.** Информатика: базовый курс : учеб. пособие для студентов вузов, бакалавров, магистров, обучающихся по направлению «Информатика и вычислит. техника», 5-е изд., М.: Омега-Л, 2008 [стр. 88-103, 107-117, 180-187].
    - **Андреева Е.В., Босова Л.Л., Фалина И.Н.** Математические основы информатики. Элективный курс, М.: БИНОМ. Лаборатория знаний, 2005 [стр. 50-96, 200-208].
    - **Савельев А.Я.** Основы информатики, М.: изд-во МГТУ им. Баумана, 2001 [стр. 63-87].
    - **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002 [стр. 501-509].
  - ◆ **Дополнительная литература**
    - **Симонович С.В.** Общая информатика, СПб.: Питер, 2007 [стр. 182-190].
    - **Борисенко В.В.** Основы программирования. *Раздел (лекция) 2 – Типы переменных. Целые и вещественные переменные, представление целых и вещественных чисел в компьютере*, Интернет-Университет информационных технологий – дистанционное образование, 2005 (<http://www.intuit.ru/department/se/pbmsu/2/>); *Раздел (лекция) 1 – Общее понятие алгоритма. Управляющие конструкции алгоритмического языка. Понятие переменной*, Интернет-Университет информационных технологий – дистанционное образование, 2005 (<http://www.intuit.ru/department/se/pbmsu/1/>).
    - **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 37-41].
    - **Казиев В.М.** Введение в информатику. *Раздел (лекция) 10 – Исполнители алгоритмов – человек и автомат*, Интернет-Университет





информационных технологий, 2006

(<http://www.intuit.ru/department/informatics/intinfo/10/>)

● **Тема (Лекции) 7-8: Примеры (решение задач и рассмотрение программ на C++). Простые переменные. Как мы подходим к решению проблемы?**

◆ Содержание лекций:

- Рассмотрение примеров (тематика лекций 3 и 4). Обсуждение.
- О книге Дьёрдя Пойа (Дж. Полия) «*How to Solve It: A New Aspect of Mathematical Method*». Последовательность действий при решении проблемы. Подход «разделяй и властвуй». Использование эвристик.
- Заключительные замечания по материалу 1-го модуля.

**окончание 1-го модуля [ 8 лекций + практические занятия ]**

● **Тема (Лекции) 9-10: Основные алгоритмические структуры. Примеры и обсуждение. Подходы к разработке алгоритмов**

◆ Содержание лекций:

- Понятие алгоритма. Исполнитель алгоритма. Свойства алгоритмов. Способы записи алгоритма (средства изображения алгоритмов – словесный, блок-схемный, псевдокод, структурные диаграммы – *сам. изучение*).
- Алгоритмы. Решение проблем с использованием компьютера (этап разработки алгоритма, этап реализации, этап сопровождения (поддержки)). Обсуждение примеров.

◆ Основная литература

- **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 15-33].
- **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 13-20].
- **Князева М.Д.** Алгоритмика: от алгоритма к программе. Учебное пособие, М.: КУДИЦ-ОБРАЗ, 2006 [стр. 5-23].
- **Давыдов В.Г.** Программирование и основы алгоритмизации, Высшая школа, 2003.
- **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 15-29].
- **Кнут Д.Э.** Искусство программирования. Том 1 (Основные алгоритмы), пер. с англ., 3-е изд., СПб.: Вильямс, 2010 [стр. 10-22 (глава «Основные понятия»)].
- **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 268-324].
- **Scheinerman E.** C++ for Mathematicians. An Introduction for Students and Professionals, Taylor & Francis Group, 2006 [стр. 31-48].
- **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002 [стр. 142-151, 521-525].

◆ Дополнительная литература

- **Акулов О.А., Медведев Н.В.** Информатика: базовый курс : учеб. пособие для студентов вузов, бакалавров, магистров, обучающихся по



направлению «Информатика и вычислит. техника», 5-е изд., М.: Омега-Л, 2008 [стр. 210-213].

- **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 39-41, 64-70].
- **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 29-32].
- **Роганов Е.А.** Практическая информатика. *Раздел (лекция) 9 – Модели и программирование*, Интернет-Университет информационных технологий, 2006 (<http://www.intuit.ru/department/se/pinform/9>).
- **Казиев В.М.** Введение в информатику. *Раздел (лекция) 7 – Базовые алгоритмические структуры*, Интернет-Университет информационных технологий, 2006 (<http://www.intuit.ru/department/informatics/intinfo/7/>).
- **Голицына О.Л., Попов И.И.** Основы алгоритмизации и программирования, 3-е изд. (доп.), М.: ИД ФОРУМ, 2008 [стр. 8-20].

● **Тема (Лекция) 11: Программирование. Что это такое? Чему и как учиться?**

(ЗАМЕЧАНИЕ: вопросы, касающиеся истории языка C++, вынесены в материалы лабораторных работ – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)

◆ **Содержание лекции:**

- Подходы к разработке алгоритмов (функциональная декомпозиция (*top-down design*) и объектно-ориентированный дизайн). Краткая характеристика подходов.
- Программный уровень. Алгоритмические языки и программирование. Области применения языков программирования. Парадигмы (вычислительные модели) программирования. Процесс трансляции (компиляторы и интерпретаторы).
- Среды проектирования<sup>2</sup> (*Integrated Development Environments*) (*сам. изучение*).
- Программирование. Чему и как учиться? Программирование – это наука, искусство, ремесло или ...? Основные понятия профессионального программирования. Правила оформления текста программ (*начало обсуждения*).

◆ **Основная литература**

- **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002 [стр. 151-171, 225-243].
- **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 6-13, 47-65].
- **Прага С.** Язык программирования C++. Лекции и упражнения, пер. с англ., 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 31-68, 85-116].
- **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 13-72].
- **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 268-324].
- **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 33-44].

<sup>2</sup> Все вопросы практического использования среды разработки MS Visual C++ 2008 (2010), написания и отладки программ вынесены на практические занятия (компьютерный практикум) курса «Основы информатики и программирования».



- ◆ **Дополнительная литература**
  - **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 71-73, 225-235].
  - **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 13-18, 25-32, 48-65].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 35-45].
  - **Одинцов И.О.** Профессиональное программирование. Системный подход, 2-е изд. (перераб.), СПб.: БХВ-Петербург, 2004 [стр. 237-249].
  - **Deitel P.J., Deitel H.M.** C++ for Programmers (Deitel® Developer Series), Prentice Hall, 2009.
  - **Одинцов И.О.** Профессиональное программирование. Системный подход, 2-е изд. (перераб.), СПб.: БХВ-Петербург, 2004 [стр. 25-72].
  - **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 45-56].
  - **Dale N.** C++ Plus Data Structures, 3<sup>rd</sup> ed., Jones and Bartlett, 2003 [стр. 1-36].
  - **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002 [стр. 510-517].
- **Тема (Лекции) 12-13: Структурное программирование, нисходящее проектирование, переход к ООП. Создание программ. Работа с данными. Выражения и преобразования. Стандартные (библиотечные) функции**

(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)
- ◆ **Содержание лекций:**
  - Основные управляющие структуры в языках программирования (операторы, выражения, подпрограммы).
  - Данные и их обработка. Структуры данных. Основные понятия структур данных. Простые (неструктурированные) типы данных. Объявления и определения. Выражения и преобразования типов. Операторы. Область действия (время жизни, модификаторы) переменных.
  - Стандартные функции. Обсуждение примеров.
- ◆ **Основная литература**
  - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 66-103].
  - **Прага С.** Язык программирования C++. Лекции и упражнения, пер. с англ., 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 116-130, 205-241, 257-285].
  - **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 13-130].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 46-67].
- ◆ **Дополнительная литература**
  - **Пахомов Б.И.** C/C++ и MS Visual C++ 2008 для начинающих, СПб.: БХВ-Петербург, 2008 [стр. 87-110].



- **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 25-32].
  - **Deitel P.J., Deitel H.M.** C++ for Programmers (Deitel® Developer Series), Prentice Hall, 2009.
- **Тема (Лекции) 14-15: Составные типы языка C++. Массивы/строки, структуры; операторы языка C++ (часть 1). Указатели, арифметика указателей (часть 1)**  
(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)
- ◆ Содержание лекции:
    - Массивы. Инициализация массивов. Многомерные массивы. Размещение массива в памяти компьютера.
    - Вложенные циклы и обработка многомерных массивов.
    - Указатели, инициализация указателей. Указатели и массивы. Арифметика указателей (адресная арифметика). Опасность указателей. Примеры (разыменование, обращение к элементам многомерных массивов).
    - Символьные массивы и строки. Обработка строк. Смешанный ввод строк и чисел. Структуры, их особенности. Массивы структур. Обсуждение примеров.
  - ◆ Основная литература
    - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 68-102].
    - **Прага С.** Язык программирования C++. Лекции и упражнения, пер. с англ., 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 131-194].
    - **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 111-162, 260-270].
    - **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 226-239].
  - ◆ Дополнительная литература
    - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 105-136].
    - **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 66-74].
    - **Пахомов Б.И.** C/C++ и MS Visual C++ 2008 для начинающих, СПб.: БХВ-Петербург, 2008 [стр. 51-56].
    - **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 268-324].
    - **Dale N.** C++ Plus Data Structures, 3<sup>rd</sup> ed., Jones and Bartlett, 2003 [стр. 588-621].
    - **Scheinerman E.** C++ for Mathematicians. An Introduction for Students and Professionals, Taylor & Francis Group, 2006 [стр. 67-78].
    - **Deitel P.J., Deitel H.M.** C++ for Programmers (Deitel® Developer Series), Prentice Hall, 2009.
- **Тема (Лекции) 16: Указатели и арифметика указателей (часть 2), свободное**



### хранилище, динамическое выделение памяти (начало)

(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)

- ◆ Содержание лекций:
  - Выделение/освобождение памяти (операторы `new` и `delete`; вариант оператора `new (nothrow)` выделения динамической памяти). Обсуждение деталей механизма выделения/освобождения памяти. Создание (размещение) динамических массивов. Обсуждение примеров.
  - Указатели и строки. Преобразование типов для указателей. Обсуждение примеров.
  - Заключительные замечания по материалу 2-го модуля.
- ◆ Основная литература
  - **Прата С.** Язык программирования C++. Лекции и упражнения, пер. с англ., 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 131-204].
  - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 111-125].
  - **Franek F.** Memory as a Programming Concept in C and C++, Cambridge University Press, 2004 [стр. 21-58, 81-105].
  - **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 111-162].
- ◆ Дополнительная литература
  - **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 70-74].
  - **Пахомов Б.И.** C/C++ и MS Visual C++ 2008 для начинающих, СПб.: БХВ-Петербург, 2008 [стр. 111-118].
  - **Scheinerman E.** C++ for Mathematicians. An Introduction for Students and Professionals, Taylor & Francis Group, 2006 [стр. 67-83].
  - **Deitel P.J., Deitel H.M.** C++ for Programmers (Deitel® Developer Series), Prentice Hall, 2009.

**окончание 2-го модуля [8 лекций + практические занятия]**

- Тема (Лекции) **17-18: Динамическое выделение памяти (продолжение).**  
**Проектирование программ. Подпрограммы (часть 1)**  
(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)
- ◆ Содержание лекций:
  - Динамическое выделение памяти, арифметика указателей. Модели памяти.
  - Парадигмы программирования (структурное программирование). Проектирование программ. Этапы проектирования и разработки программ.
  - Понятие подпрограммы (процедуры/функции). Подпрограммы как частный случай сопрограмм. Функции: программные модули C++. Прототип и вызов функции. Механизм вызова функции. Перегрузка функций.
- ◆ Основная литература
  - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 137-150, 152-157, 180-188].





- **Прага С.** Язык программирования C++. Лекции и упражнения, 5-е изд., М.: ООО «И.Д.Вильямс», 2007 [стр. 159-167, 303-349].
- **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 150-162, 170-191].
- ◆ **Дополнительная литература**
  - **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 33-36, 40-47, 76-81].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 121-163].
  - **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 251-258].
  - **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 286-297].
  - **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 268-324].

● **Тема (Лекции) 19-20: Подпрограммы (часть 2). Ссылки (ссылочные переменные)**

(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)

- ◆ **Содержание лекций:**
  - Прототип и вызов функции. Передача параметров (аргументов) по значению и по ссылке. Шаблоны (*templates*) функций. Функции и массивы. Функции и строки. Функции и структуры. Указатели на функции. Обсуждение примеров.
  - Ссылочные переменные (*references*). Ссылки как параметры функций.
  - Функции, возвращающие ссылки. Ссылки и указатели. Примеры программ (объяснение и обсуждение).
- ◆ **Основная литература**
  - **Прага С.** Язык программирования C++. Лекции и упражнения, 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 307-362].
  - **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 196-238].
  - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 137-150, 152-157].
  - **Deitel Н.М., Deitel Р.Ж.** C++ How to Program, 6<sup>th</sup> ed., Pearson Educ. (Prentice Hall), 2008 [стр. 281-286].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 128-163].
  - **Тимофеев В.В.** Самоучитель C++ (C++ как он есть), М.: БИНОМ, 2009 [стр. 140-144].
  - **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 254-264].
- ◆ **Дополнительная литература**
  - **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 33-36, 40-47].





- **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 286-297].

- **Тема (Лекции) 21-22: Рекурсия**

(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)

- ◆ **Содержание лекций:**

- Рекурсивный вызов подпрограмм. Рекурсия с одиночным и множественными вызовами. Сильные и слабые стороны рекурсии. Итеративный (основанный на циклах) алгоритм vs. рекурсивный алгоритм. Механизм рекурсии (запись активации, ее структура).
- Обсуждение примеров (вычисление чисел Фибоначчи, факториала, сортировка и поиск в массиве).

- ◆ **Основная литература**

- **Прага С.** Язык программирования C++. Лекции и упражнения, 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 349-354].
- **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 192-196].
- **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 157-160].
- **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 94-99].
- **Dale N.** C++ Plus Data Structures, 3<sup>rd</sup> ed., Jones and Bartlett, 2003 [стр. 399-436].

- ◆ **Дополнительная литература**

- **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 36-39].
- **Langsam Y., Augenstein M.J.** Data Structures Using C and C++, 2<sup>nd</sup> ed., Prentice Hall, 1995 [стр. 117-170].
- **Franek F.** Memory as a Programming Concept in C and C++, Cambridge University Press, 2004 [стр. 59-80].

- **Тема (Лекции) 23-24: Дополнительные примеры (лекции 19-22). Препроцессор**

- ◆ **Содержание лекции:**

- Обсуждение дополнительных примеров по разным темам дисциплины.
- Препроцессор. Использование директив препроцессора.

- ◆ **Основная литература**

- **Прага С.** Язык программирования C++. Лекции и упражнения, 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 349-354].
- **Подбельский В.В.** Стандартный C++, М.: Финансы и статистика, 2008 [стр. 192-196].
- **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 157-160].
- **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 94-99].
- **Dale N.** C++ Plus Data Structures, 3<sup>rd</sup> ed., Jones and Bartlett, 2003 [стр. 399-436].



- ◆ **Дополнительная литература**
  - **Баженова И.Ю., Сухомлин В.А.** Введение в программирование, М.: Интернет-Университет информационных технологий, БИНОМ. Лаборатория знаний, 2007 [стр. 36-39].
  - **Langsam Y., Augenstein M.J.** Data Structures Using C and C++, 2<sup>nd</sup> ed., Prentice Hall, 1995 [стр. 117-170].
  - **Franek F.** Memory as a Programming Concept in C and C++, Cambridge University Press, 2004 [стр. 59-80].
- **Тема (Лекция) 25-26: Линейные структуры данных: стек и очередь как примеры абстрактных типов данных (АТД)**
- ◆ **Содержание лекции:**
  - Алгоритмы, ориентированные на структуры данных. Понятие связного списка (*linked list*). Операции, выполняемые над структурами данных. Работа со связным списком (создание, добавление/удаление узлов списка, и т.п.).
  - Линейные и нелинейные структуры данных. Классификация линейных структур данных (стек (*stack*) и очередь (*queue*)). Реализация линейных структур. Примеры (инфиксная, постфиксная и префиксная формы записи арифметических выражений; перевод «инфикс–постфикс», вычисление постфиксных выражений и пр.).
  - Контейнеры (*stack, queue*) (*сам. изучение*).
  - [опция] Процедурная абстракция. Понятие интерфейса и реализации. Обсуждение примера (сортировка). Спецификации функции (предусловие и постусловие).
- ◆ **Основная литература**
  - **Hubbard J.R.** Schaum's Outline of Programming with C++, 2<sup>nd</sup> ed., McGraw-Hill Publ., 2000 [стр. 156-182].
  - **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 5-15].
  - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [стр. 315-330].
  - **Hubbard J.R.** Schaum's Outline of Theory and Problems of Fundamentals of Computing with C++, McGraw-Hill Publ., 1998 [стр. 137-140].
  - **Deitel H.M., Deitel P.J.** C++ How to Program, 5<sup>th</sup> ed., Pearson Educ. (Prentice Hall), 2005 [глава 6, части 6.4-6.7].
  - **Eckel B.** Thinking in C++, 2<sup>nd</sup> ed., Prentice Hall, 2000 [глава 1 (Introduction to Objects), стр. 29-34].
  - **Скляров В.А.** Язык C++ и объектно-ориентированное программирование, Минск - Высшая Школа, 1997 [стр. 10-43].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 128-173].
  - **Hubbard J.R.** Schaum's Outline of Programming with C++, 2<sup>nd</sup> ed., McGraw-Hill Publ., 2000 [стр. 87-94].
  - **Scheinerman E.** C++ for Mathematicians. An Introduction for Students and Professionals, Taylor & Francis Group, 2006 [стр. 144-152].
  - **Dale N.** C++ Plus Data Structures, 3<sup>rd</sup> ed., Jones and Bartlett, 2003 [стр. 195-201].
  - **Липпман С., Лажоие Ж., Му Б.** Язык программирования C++. Вводный курс



(пер. с англ.), Вильямс, 2007 [главы 9 и 10].

- **Brookshear J.G.** Computer Science. An overview, 9<sup>th</sup> ed., Pearson Educ., 2007 [стр. 349-394].
- ◆ **Дополнительная литература**
  - **Фридман А.Л.** [Распределение памяти](#) (лекция №9). Лекция из курса «Язык программирования С++», Интернет-Университет информационных технологий – дистанционное образование, 2003.
  - **Могилев А.В., Пак Н.И., Хеннер Е.К.** Информатика: учеб. пособие для студентов пед. вузов, 5-е изд., М.: Издательский центр «Академия», 2007 [стр. 78-82].
  - **Langsam Y., Augenstein M.J.** Data Structures Using C and C++, 2<sup>nd</sup> ed., Prentice Hall, 1995 [стр. 77-114].
  - **Мейер Б.** Материалы лекции из курса «Объектно-ориентированное программирование» - [Три возможных варианта представления стека](#), 2006.
  - **Мейер Б.** [Абстрактные типы данных](#) (лекция №6). Лекция из курса «[Основы объектно-ориентированного программирования](#)», Интернет Университет Информационных Технологий, [INTUIT.ru](#), 2006.

**окончание 3-го модуля [ 10 лекций + практические занятия ]**

● **Тема (Лекция) 27-28: Абстракция данных. Понятия класса и объекта**

- ◆ **Содержание лекции:**
  - Абстракция данных. Использование С++ как объектно-ориентированного языка. Что такое тип? Понятие класса. Описание класса. Конструктор класса. Объект класса. Функции-члены класса. Отделение определения функций от их объявления. Примеры классов.
  - Абстрактный тип данных (ADT) как описание «идеального» типа. Процедурное мышление. Правила С++-проектирования. Использование объектов. Объектно-ориентированная модель.
- ◆ **Основная литература**
  - **Hubbard J.R.** Schaum's Outline of Theory and Problems of Fundamentals of Computing with C++, McGraw-Hill Publ., 1998 [стр. 137-154].
  - **Deitel H.M., Deitel P.J.** C++ How to Program, 5<sup>th</sup> ed., Pearson Educ. (Prentice Hall), 2005 [глава 3, части 3.1-3.9].
  - **Скляров В.А.** Язык С++ и объектно-ориентированное программирование, Минск - Высшая Школа, 1997 [стр. 10-43].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 128-173].
  - **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 277-286].
  - **Шилдт Г.** Самоучитель С++ (пер. с англ.), БНВ-Петербург, 3-е изд., 2006 [стр. 26-32].
  - **Солтер Н., Клепер С.** С++ для профессионалов (пер. с англ.), Диалектика, 2006 [стр. 54-76, 85-105].
- ◆ **Дополнительная литература**
  - **Страуструп Б.** [Абстракция данных в языке С++](#) (CyberGuru.ru, 2006-2008).
  - **Мейер Б.** [Статические структуры: классы](#) (лекция №7), [динамические](#)



[структуры: объекты](#) (лекция №8). Лекции из курса [«Основы объектно-ориентированного программирования»](#), Интернет Университет Информационных Технологий, [INTUIT.ru](#), 2006.

- **Фридман А.Л.** [Шаблоны](#) (лекция №18). Лекция из курса «Язык программирования С++», Интернет-Университет информационных технологий – дистанционное образование, 2003.
- **Deitel P.J., Deitel H.M.** C++ for Programmers (Deitel® Developer Series), Prentice Hall, 2009.

● **Тема (Лекция) 29: Принципы объектно-ориентированного программирования (ООП)**

(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)

◆ Содержание лекции:

- Процедурное и объектно-ориентированное программирование. Абстракция и классы. Проектирование, нацеленное на многократное использование кода (*code reuse*). Стандартная библиотека С++. Этапы разработки программных продуктов с использованием ООП. Иерархия, полиморфизм, инкасуляция (общее обсуждение, терминология).
- Объекты программы и объекты реального мира. Терминология ООП.
- Класс с деструктором. Обзор конструкторов и деструкторов (*начало обсуждения*).

◆ Основная литература

- **Хабибуллин И.Ш.** Программирование на языке высокого уровня. С/С++, СПб.: БХВ-Петербург, 2006 [стр. 223-267].
- **Прага С.** Язык программирования С++. Лекции и упражнения, 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [стр. 483-534].
- **Довбуш Г.Ф., Хомоненко А.Д.** Visual С++ на примерах, СПб.: БХВ-Петербург, 2008 [стр. 175-189].
- **Пахомов Б.И.** С/С++ и MS Visual С++ 2008 для начинающих, СПб.: БХВ-Петербург, 2008 [стр. 155-173].
- **Hubbard J.R.** Schaum's Outline of Programming with С++, 2<sup>nd</sup> ed., McGraw-Hill Publ., 2000 [стр. 232-243].
- **Шилдт Г.** Самоучитель С++, ВHV-Петербург, 3-е изд., 2006 [стр. 43-57].
- **Колдаев В.Д.** Основы алгоритмизации и программирования: учебное пособие, М.: ИД ФОРУМ, Инфра-М, 2009 [стр. 287-289].
- **Тимофеев В.В.** Самоучитель С++ (С++ как он есть), М.: БИНОМ, 2009 [стр. 155-162].
- **Dale N., Lewis J.** Computer Science Illuminated, Jones and Bartlett Publ., 2002 [стр. 162-175, 261-264].
- **Солтер Н., Клепер С.** С++ для профессионалов (пер. с англ.), Диалектика, 2006 [стр. 138-154].

◆ Дополнительная литература

- [Процедурное программирование – Объектно-ориентированное программирование](#) (материал из Википедии, 2008).
- **Мейер Б.** [Модульность](#) (лекция №3). Лекция из курса [«Основы объектно-ориентированного программирования»](#), Интернет Университет Информационных Технологий, [INTUIT.ru](#), 2006.



- **Фридман А.Л.** [Классы – конструкторы и деструкторы](#) (лекция №12). Лекция из курса «Язык программирования С++», Интернет-Университет информационных технологий – дистанционное образование, 2003.
- **Тема (Лекции) 30-31: Классы и объекты. Методы класса. Указатели на объекты**  
(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)
- ◆ **Содержание лекций:**
  - Краткое ознакомление с UML. Диаграмма классов (*class diagram*) (*сам. изучение*).
  - Создание классов. Методы класса. Изменение реализации. Создание и разрушение объектов. Конструктор копирования. Отличие копирования от присваивания. Классы и динамическое выделение памяти (динамическое выделение памяти в объектах) (*начало*).
  - Указатели на объекты. Указатель `this`. Примеры и обсуждение (*начало*).
- ◆ **Основная литература**
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [*стр. 182-203*].
  - **Хабибуллин И.Ш.** Программирование на языке высокого уровня. C/C++, СПб.: БХВ-Петербург, 2006 [*стр. 253-267*].
  - **Лафоре Р.** Объектно-ориентированное программирование в C++, 4-е изд., СПб.-Питер, 2007 [*стр. 217-260*].
  - **Прата С.** Язык программирования C++. Лекции и упражнения, 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [*стр. 483-520, 591-656*].
  - **Шилдт Г.** C++ для начинающих (пер. с англ.), Эком, 2007 [*модуль 8*].
  - **Ашарина И.В.** Объектно-ориентированное программирование в C++: Лекции и упражнения, М.: Горячая Линия – Телеком, 2008 [*стр. 192-209*].
  - **Тимофеев В.В.** Самоучитель C++ (C++ как он есть), М.: БИНОМ, 2009 [*стр. 163-172*].
- ◆ **Дополнительная литература**
  - **Фридман А.Л.** [Классы и объекты](#) (лекция №7). Лекция из курса «Язык программирования С++», Интернет-Университет информационных технологий – дистанционное образование, 2003.
  - [Введение в UML. Диаграмма классов](#) / Проинфотех, 2003.
  - **Леоненков А.** [Самоучитель UML – Введение \(методология процедурно-ориентированного программирования \(ПОП\) / методология ООП\) – Класс \(секция 5.1\), объекты \(секция 5.4\)](#).
  - [UML - новый стандарт языка объектно-ориентированного моделирования. Квинтэссенция успешного опыта](#), Interface.ru (Internet & Software Company), 2006.
  - **Deitel P.J., Deitel H.M.** C++ for Programmers (Deitel® Developer Series), Prentice Hall, 2009.
  - **Franek F.** Memory as a Programming Concept in C and C++, Cambridge University Press, 2004 [*стр. 106-131*].
- **Тема (Лекции) 32-36: Классы и объекты. Классы и динамическое выделение памяти (окончание). Класс string. Дополнительные примеры**





(ЗАМЕЧАНИЕ: большая часть обсуждения вынесена в материалы для практических занятий – т.н. *prelab activity* – см. ДОПОЛНИТЕЛЬНЫЕ ФАЙЛЫ)

- ◆ Содержание лекций:
  - Статические данные и статические методы класса. Константные данные и константные методы класса. Ссылочные члены данных. Комментарии и примеры (*окончание*).
  - Использование `new` в конструкторах.
  - Возвращаемые объекты. Массивы объектов. Указатели на объекты (*окончание*). Обсуждение примеров.
  - Класс `string` (*сам. изучение*).
  - Сессия «вопросы-ответы»: обсуждение вопросов, касающихся рассмотренных в курсе тем (заключительная часть модуля и дисциплины в целом); рекомендации по дальнейшему изучению материала.
- ◆ Основная литература
  - **Прага С.** Язык программирования C++. Лекции и упражнения, 6-е изд., М.: ООО «И.Д.Вильямс», 2012 [*стр. 535-549*].
  - **Довбуш Г.Ф., Хомоненко А.Д.** Visual C++ на примерах, СПб.: БХВ-Петербург, 2008 [*стр. 256-292*].
  - **Deitel Н.М., Deitel Р.Ж.** C++ How to Program, 6<sup>th</sup> ed., Pearson Educ. (Prentice Hall), 2008 [*стр. 578-623*].
  - **Лафоре Р.** Объектно-ориентированное программирование в C++, 4-е изд., СПб.-Питер, 2007 [*стр. 312-260*].
  - **Шилдт Г.** Самоучитель C++ (пер. с англ.), БНВ-Петербург, 3-е изд., 2006 [*глава 6, стр. 175-203*].
  - **Шилдт Г.** C++ для начинающих (пер. с англ.), Эком, 2007 [*модуль 9*].
  - **Тимофеев В.В.** Самоучитель C++ (C++ как он есть), М.: БИНОМ, 2009 [*стр. 176-183*].
- ◆ Дополнительная литература
  - Интернет-ресурс: [Введение в ООП – Указатель this](#).

## 8 Образовательные технологии

Практические занятия первого модуля дисциплины предусматривают подробное объяснение каждой разбираемой программы, совместную работу (в параллель) со студентами, акцентирование внимания студентов на наиболее значимых аспектах кода.

Кроме этого, в курсе активно используются материалы (файлы) для самостоятельной работы студентов – эти файлы включают

- *объяснения*, связанные с материалом предстоящих практических занятий (в компьютерном классе) – очередной файл из состава КМ размещается в LMS, по крайней мере, за 8-12 дней до очередного практического занятия, что дает возможность студентам внимательно изучить материалы, проконсультироваться с преподавателем и быть готовым к соответствующим занятиям,
- *примеры* C++ программ с детальными комментариями, графическими пояснениями, схемами, результатами тестовых прогонов (скриншотами) и т.п.,
- *ссылки* на русскоязычные и англоязычные интернет-ресурсы, дающие студентам дополнительные полезные сведения и объяснения в процессе работы с учебными материалами,





- *комментарии и советы* экспертов (в частности, Б.Страуструпа, А.Левитина, А.Хортон, Г.Шилдта, С.Мейерса, Н.Вирта, С.Дьюхэрста, Р.Пайка, П.Гудлифа, Б.Пахомова, и др.), дающие студентам хорошую возможность поразмышлять и сформировать свое мнение по соответствующему вопросу,
- текущее *домашнее задание* как одна из составляющих самостоятельной работы студентов (разработка и написание от 2 до 5 программ в качестве дополнительного практикума),
- *объяснение* ключевых терминов (определения), ссылки на рекомендуемую и использованную при подготовке КМ литературу,
- *вопросы* для самоконтроля.

Для удобства использования и придания процессу обучения последовательного характера (этапности) КМ разбит на отдельные файлы, длина которых варьируется от 10 до 26 страниц (средний размер файла составляет  $\approx$  18-19 страниц). В результате, перед каждым занятием студенты четко знают тему (и приблизительное содержание) их практического задания, которое предстоит выполнить в течение 4-х академических часов практикума.

## 9 Оценочные средства для текущего контроля и аттестации студента

### 9.1 Тематика заданий текущего контроля

#### Контрольные работы (КР)

Подробное объяснение форм контроля, применяемого в курсе, представлено на стр.8-9 данного документа. В курсе запланированы три контрольные работы, которые имеют практическую направленность – все они проводятся в компьютерных классах в **конце 2, 3 и 4 модулей**. Студентам предлагаются задания, связанные с обсуждаемым в течение соответствующих учебных модулей в курсе материалом. Целью контрольных работ является проверка знаний и навыков студентов, связанных с разработкой, написанием и отладкой программ на языке C++. Продолжительность каждой КР – от 120 до 140 минут, в зависимости от сложности предлагаемого(-ых) задания(-й).

#### Тематика вопросов контрольных работ (некоторые общие варианты):

- задания, связанные с переводом чисел из одной р-ичной системы в другую,
- задания, связанные с выявлением синтаксических ошибок в приведенном варианте (фрагменте) программы,
- задания, связанные с проверкой понимания студентами синтаксиса основных операторов языка программирования,
- задания, связанные с проверкой понимания студентами правил выполнения операторов цикла (вложенных циклов),
- задания, связанные с определением области действия переменных в подпрограммах (функциях), рассмотрением механизмов передачи параметров (аргументов) в подпрограммы (функции) – в широком смысле, вопросы по использованию пользовательских (*user-defined*) функций,
- задания, связанные с проверкой умения студентов выбрать подходящую структуру данных для решения поставленной задачи,
- задания, связанные с разработкой алгоритма (псевдокода) и написания программы решения задачи по обработке массивов/строк,
- задания, связанные с проверкой понимания студентами рекурсивных алгоритмов, умения трассировать рекурсивные алгоритмы (программы),
- задания, связанные с разработкой класса(-ов) и их реализацией в соответствии с постав-



ленной задачей.

## 9.2 Вопросы для оценки качества освоения дисциплины (примеры)

Достаточно обширный перечень потенциальных вопросов может включать, в частности, следующие контрольные вопросы:

- Какие определения информатики вы знаете?
- Какова общая структура современной информатики?
- Какое место занимает информатика в системе наук (как информатика связана с другими науками)?
- Какие негативные последствия несет с собой появление новых информационных технологий?
- Можно ли сказать, что компьютер представляет собой программно-управляемый цифровой автомат?
- В чем отличие позиционной системы счисления от непозиционной?
- Как определяется алфавит? Что называется основанием системы счисления?
- Каковы способы перевода чисел из одной системы счисления в другую?
- Почему при работе с компьютерами используются восьмеричная (*oct*) и шестнадцатеричная (*hex*) системы счисления?
- Как представляются вещественные числа в компьютере? Что такое «нормализованная запись числа»? Как представляются символьные данные в компьютере?
- Кратко объясните формальные правила двоичной арифметики.
- Определите, кто или что называется исполнителем алгоритма.
- Что такое математическая индукция?
- Сформулируйте последовательность действий при решении проблемы (следуя книге Дьёрдя Пойа «*How to Solve It ...*»).
- Охарактеризуйте базовые структуры алгоритмов.
- В чем состоит модульность при структурной разработке алгоритмов?
- Назовите основные свойства алгоритма.
- Кратко охарактеризуйте способы записи алгоритмов.
- Что Вы можете сказать о подходе функциональной декомпозиции (*top-down design*) к разработке алгоритмов.
- Что собой представляет порядок (шаги) создания программы?
- Какие данные можно отнести к простейшим неструктурированным?
- В чем заключается процесс трансляции программы? В чем заключается отличие транслятора от интерпретатора?
- Сформулируйте основные понятия профессионального программирования.
- Опишите этапы проектирования и разработки программ.
- Объясните, что такое тип. Как можно определить понятие "переменная"?
- Для чего необходимо объявлять переменные?
- Что такое идентификатор?
- Что собой представляет тип `void`?
- Приведите примеры *escape*-последовательностей?
- Кратко объясните классификацию структур данных по признаку изменчивости.
- Что такое массив? Охарактеризуйте свойства данных целого, действительного типа.
- Что такое преобразование типов (в языке программирования)?
- Какова цель использования директивы `using`? Каков смысл использования директивы `using namespace std`?
- Как Вы понимаете словосочетание «абстрактный тип данных» (АТД/ADT)?



- Какие вы знаете основные структуры управления языка C++?
- Какими правилами определяется очередность выполнения операций в выражении?
- В чём заключается особенность операции деления применительно к целочисленным операндов?
- Что такое автоматическое (*implicit*) приведение (преобразование) типов?
- Что такое целочисленный литерал?
- Какие операторы могут прервать выполнение цикла до его завершения, определяемого выражением-условием?
- Каково минимальное число итераций в цикле с постусловием?
- Каково назначение оператора `break`. Где его можно применять?
- В каких ситуациях уместно (целесообразно) использовать оператор `goto`?
- Каково назначение оператора `continue`?
- Что такое указатель? Сформулируйте правила адресной арифметики. Возможно ли применение операторов `++` и `--` к переменным-указателям?
- Что собой представляет постфиксная (инфиксная) форма записи арифметического выражения?
- Что называется прототипом функции? Может ли функция возвращать несколько значений?
- Верно ли утверждение: «Присутствие точки с запятой в прототипе функции идентифицирует его как оператор»?
- Передача указателей функциям. Использование кучи (*heap*) – что называется кучей?
- Как выполняется бинарный поиск в массиве (кратко опишите процедуру)?
- В чем отличие ссылки от указателя? Может ли функция возвращать ссылку? Есть ли в этом какая-либо опасность?
- Приведите пример функции, возвращающей ссылку. Кратко поясните смысл возвращаемого значения.
- Как создать динамический одномерный (двумерный) массив в C++ программе?
- Как нумеруются элементы массива? Каков смысл индекса элемента массива?
- Что собой представляет односвязный список? Объясните, каким образом выполняются операции (1) добавления нового элемента (узла) в конец списка, (2) удаление заданного элемента, (3) удаления первого элемента (узла) списка?
- Что собой представляют строки в стиле C? Какие вы знаете стандартные библиотечные функции, предназначенные для их обработки? Каковы преимущества использования стандартного класса (типа) `string`?
- Можно ли объект типа (класса) `string` инициализировать строкой в стиле C? Приведите пример(ы) и дайте объяснение.
- Что собой представляет списковая инициализация строк (в стиле C) и объектов `string` (C++11)?
- В чем заключается особенность использования варианта оператора `new(nothrow)` выделения динамической памяти?
- Что такое рекурсия? В чем, на ваш взгляд, проявляются сильные и слабые стороны рекурсии?
- Какие вы знаете парадигмы программирования?
- Кратко сформулируйте и объясните принципы C++-проектирования.
- Что такое класс и объект?
- Что такое структура (тип `struct`)?
- Что такое дескриптор структуры?
- Что такое «конструктор класса»? Может ли класс содержать несколько конструкторов?
- Сколько параметров имеет конструктор копии (копирования)? В чем отличие копирова-



ния от присваивания?

- Может ли в классе отсутствовать явно определенный деструктор?
- В чем специфика статических методов (элементов-функций) класса?
- Что такое стек? Что такое очередь?

## 10 Учебное и информационное обеспечение дисциплины

### 10.1 Программные средства

При проведении практических занятий дисциплины используется программная среда Microsoft Visual Studio 2008 (2010) или Microsoft Visual C++ 2008 (2010) Express Edition, установленные в большинстве компьютерных классах. Как было отмечено выше, для самостоятельной работы (на собственных компьютерах) студенты могут использовать и другие C++ компиляторы (среды разработки), напр.,

- **Digital Mars C/C++** (<http://www.digitalmars.com/download/freecompiler.html>),
- **MinGW** (<http://www.mingw.org/>),
- **Code::Blocks** (<http://www.codeblocks.org/>),
- **Orwell Dev-C++** (<http://orwelldevcpp.blogspot.ru/>),
- **XCode** (Apple's IDE; <https://developer.apple.com/xcode/>) и др.

### 10.2 Дистанционная поддержка дисциплины

Все материалы дисциплины, объявления, текущие оценки и другие материалы, включая *"Методику формирования оценки за программную реализацию задачи"*, размещаются на портале LMS (информационной образовательной среды НИУ ВШЭ, <http://lms.hse.ru/>).

Автор программы: \_\_\_\_\_ Дегтярев К.Ю.