

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное агентство по образованию

Государственное образовательное учреждение
высшего профессионального образования
«Оренбургский государственный университет»

Кафедра программного обеспечения вычислительной техники
и автоматизированных систем

УТВЕРЖДАЮ
Декан факультета информационных технологий
_____ А.М. Пищухин
« ____ » _____ 2006 г.

Рабочая программа

дисциплины «Объектно-ориентированное программирование на языке C++»

Направление подготовки: 654600 – Информатика и вычислительная техника
Специальность: 220400 “Программное обеспечение вычислительной техники и автоматизированных систем ”
Факультет информационных технологий
Форма обучения: заочная

Оренбург 2006

Рецензент: Соловьев д.т.н., профессор

Рабочая программа дисциплины «Объектно-ориентированное программирование на языке С++» /сост. Я.М. Коновал – Оренбург: ОГУ, 2006 - 26 с.

Предназначена для преподавания специальной дисциплины студентам специальности 220400 в 7-м семестре для заочной формы обучения.

Рабочая программа составлена с учетом Государственного образовательного стандарта высшего профессионального образования по направлению подготовки дипломированных специалистов - 654600 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА (утвержденного 27.03.2000 Министерством Образования Российской Федерации).

Составитель: _____ Я. М. Коновал
22.05.2006 г.

Содержание

1	Цели и задачи дисциплины	4
2	Место дисциплины в учебном процессе	5
3	Организационно-методические данные дисциплины	6
4	Содержание дисциплины	7
	4.1 Разделы дисциплины, их содержание и виды занятий	7
5	Тематический план изучения дисциплины	9
	5.1 Лекции.....	9
	5.2 Лабораторные работы	11
	5.3 Самостоятельное изучение разделов дисциплины	12
6	Учебно-методическое обеспечение дисциплины	13
6.1	Рекомендуемая литература	13
6.1.1	Основная литература	13
6.1.2	Дополнительная литература	13
7	Материально-техническое обеспечение дисциплины	14
	7.1 Учебно-лабораторное оборудование	14
	Лист согласования рабочей программы	15

1 Цели и задачи дисциплины

В результате изучения дисциплины студент должен сформировать минимально-необходимый комплекс знаний и умений:

а) иметь представление:

- 1) об общих принципах работы трансляторов и редакторов;
- 2) об общих принципах работы с системами Borland C++ Builder и Microsoft Visual C++;
- 3) об основных принципах объектно-ориентированного программирования;
- 4) об основных принципах разработки эффективных программ;
- 5) о методах разработки сложных программных комплексах;
- 6) о преимуществах и недостатках объектно-ориентированных языков программирования;
- 7) методах отладки программных комплексов;
- 8) о тенденциях и перспективах развития современных средств программирования.

б) знать:

- 1) структуру программ на языке C++;
- 2) синтаксис и семантику основных операторов языка C++;
- 3) основные идеи объектно-ориентированного программирования;
- 4) способы создания и использования классов и объектов;
- 5) основные идеи и примеры использования инкапсуляции;
- 6) основные идеи и примеры использования наследования;
- 7) принцип полиморфизма и использование виртуальных функций;
- 8) создание шаблонов функций и классов;
- 9) механизмы создания и обработки исключений.

в) должен владеть:

- 1) методами анализа эффективности алгоритмов решения задач;
- 2) средствами создания сложных программных комплексов;
- 3) методами выбора структуры используемых переменных и методов для создания классов;
- 4) средствами разработки программного обеспечения.

2. Место дисциплины в учебном процессе

Дисциплина относится к циклу общепрофессиональных дисциплин федерального компонента основной образовательной программы подготовки инженера по специальности 230105.65 «Программное обеспечение вычислительной техники и автоматизированных систем».

Дисциплина базируется на следующих дисциплинах: «Программирование на языке высокого уровня»; «Организация ЭВМ и систем»; «Операционные системы».

Знания, умения и навыки, приобретенные при изучении данной дисциплины используются при разработке дипломного проекта.

3. Организационно-методические данные дисциплины

Вид работы	Трудоемкость, ч
Общая трудоемкость	152
Аудиторная работа	72
лекции (Л)	36
лабораторные работы (ЛР)	36
Самостоятельная работа	80
Расчетно-графическое задание (РГЗ)	36
Самоподготовка (самостоятельное изучение разделов, проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к лабораторным работам и т.д.), (СР)	44
Вид итогового контроля	Зачет, экзамен

4. Содержание дисциплины

Дисциплина изучается в седьмом семестре

4.1 Разделы дисциплины, виды занятий и характеристика учебной деятельности студентов

№ раздела	Ссылки на цели	Наименования разделов	Количество часов			Внеауд. работа СР	
			Всего	Аудиторная Работа			
				Л	ПЗ		ЛР
1	1, 2, 4, 8	Раздел 1. С++ как расширение языка С. пространства имен.	12	4		8	
2	2, 3, 4, 9, 10, 15, 16	Раздел 2. Основы С++	24	4		8	12
3	2, 3, 4, 7, 11, 15, 16	Раздел 3. Наследование в С++	26	6		8	12
4	2, 3, 4, 7, 12, 15, 16	Раздел 4. Обработка исключений в С++	20	4		4	12
5	2, 3, 4, 7, 14, 15, 17	Раздел 5. Потоки ввода-вывода в С++	26	6		8	12
6	2, 3, 4, 7, 13, 15	Раздел 6. Шаблоны в С++	26	6		8	12
7	2, 3, 4, 6, 14, 15, 17	Раздел 7 Библиотека шаблонов STL.	18	6			12
		<i>Итого:</i>	152	36		36	80

4.2 Содержание разделов дисциплины

Раздел 1. Начальное изучение C++. C++ как расширение языка C.

ужесточающие:

обязательное использование прототипов функций перед их использованием;
обязательное указание явного преобразования типа для типа void;

улучшающие:

дополнительный вид комментариев;
использование в функциях аргументов по умолчанию;
использование ссылок как псевдонимов-алиасов;
использование ссылок в параметрах функций и при возвращении результатов;

встроенные функции;

разрешение видимости глобальных переменных с помощью :: ;
использование перегружаемых функций;
гибкое определение переменных в любом месте кода ;
константы;
гибкие операторы распределения памяти;
присваивание структур, объединений и множеств;
отсутствие необходимости употреблять в определениях структур, объединений и множеств соответствующих ключевых слов;
пространства имен.

Раздел 2. Основы C++

Краткий обзор. Функции-члены и данные-члены. Интерфейсы и реализация. Конструкторы и деструкторы. Инициализация и очистка. Конструктор без параметров (по умолчанию). Конструктор копирования. Указатель this.

Статические члены: функции и данные. Указатели на члены. Структуры и объединения. Константные члены-функции и константные объекты.

Функции-друзья.

Перегрузка бинарных и унарных операций. Перегруженные операции индексирования, вызова функций, инкремента и декремента префиксных и постфиксных, разадресации. Перегрузка new, delete.

Преобразование типов, определяемых пользователем с помощью конструкторов и операций преобразования. Неявное преобразование типов. Друзья-функции и друзья-классы.

Раздел 3. Наследование в C++

Наследование классов и производные классы. Конструкторы, деструкторы и наследование.

Иерархия классов. Виртуальные функции. Полиморфизм. Абстрактные классы и чистые виртуальные функции. Множественное наследование. Виртуальные базовые классы. Контроль доступа.

Определение типа объектов во время выполнения программы (RTTI).

Раздел 4. Обработка исключений в C++

Обработка ошибок в стандартном C. Распознавание ситуаций. Использование assert, кодов возврата, сигналов, setjmp и longjmp. Использование set_new_handler(). Объектно-ориентированная обработка исключений. Применение try, catch, throw.

Раскрутка стека. Стандартные исключения в C++. Работа с конструкторами и исключениями. Функции terminate(), unexpected().

Раздел 5. Поток ввода-вывода в C++

Заголовочные файлы. Предопределенные объекты и потоки.

Операции помещения и извлечения. Форматирование. Флаги форматирования. Манипуляторы.

Ошибки потоков. Файловый ввод-вывод с применением потоков C++.

Конструкторы файловых потоков. Открытие файлов в разных режимах. Ввод-вывод в файлы.

Форматирование в памяти.

Раздел 6. Шаблоны в C++

Шаблоны функций. Шаблоны классов. Параметры шаблонов. Наследование и шаблоны. Примеры построения шаблонов. Итераторы.

Раздел 7

Библиотека шаблонов STL.

Векторы, списки и очереди. Контейнерные классы. Используемые алгоритмы.

5 Тематический план изучения дисциплины

5.1 Лекции

Лекции, читаемые в 7 семестре

№ раз-дела	Ссылка на цели	Наименования, тем	Кол-во часов
1	1, 2, 4, 8	<p>Раздел 1. Начальное изучение C++. C++ как расширение языка C.</p> <p>ужесточающие:</p> <ul style="list-style-type: none"> обязательное использование прототипов функций перед их использованием; обязательное указание явного преобразования типа для типа void; <p>улучшающие:</p> <ul style="list-style-type: none"> дополнительный вид комментариев; использование в функциях аргументов по умолчанию; использование ссылок как псевдонимов-алиасов; использование ссылок в параметрах функций и при возвращении результатов; встроенные функции; разрешение видимости глобальных переменных с помощью :: ; использование перегружаемых функций; гибкое определение переменных в любом месте кода ; константы; гибкие операторы распределения памяти; присваивание структур, объединений и множеств; отсутствие необходимости употреблять в определениях структур, объединений и множеств соответствующих ключевых слов; пространства имен. 	4
2	2, 3, 4, 9, 10, 15, 16	<p>Раздел 2. Основы C++</p> <p>Краткий обзор. Функции-члены и данные-члены. Интерфейсы и реализация. Конструкторы и деструкторы. Инициализация и очистка. Конструктор без параметров (по умолчанию). Конструктор копирования. Указатель this.</p> <p>Статические члены: функции и данные. Указатели на члены. Структуры и объединения. Константные члены-функции и константные объекты.</p> <p>Функции-друзья.</p> <p>Перегрузка бинарных и унарных операций.</p> <p>Перегруженные операции индексирования, вызова функций, инкремента и декремента префиксных и постфиксных, разадресации. Перегрузка new, delete.</p> <p>Преобразование типов, определяемых пользователем с помощью конструкторов и операций преобразования. Неявное преобразование типов. Друзья-функции и друзья-классы.</p>	4

№ раз-дела	Ссылка на цели	Наименования, тем	Кол-во часов
3	2, 3, 4, 7, 11, 15, 16	Раздел 3. Наследование в C++ Наследование классов и производные классы. Конструкторы, деструкторы и наследование. Иерархия классов. Виртуальные функции. Полиморфизм. Абстрактные классы и чистые виртуальные функции. Множественное наследование. Виртуальные базовые классы. Контроль доступа. Определение типа объектов во время выполнения программы (RTTI).	6
4	2, 3, 4, 7, 12, 15, 16	Раздел 4. Обработка исключений в C++ Обработка ошибок в стандартном C. Распознавание ситуаций. Использование assert, кодов возврата, сигналов, setjmp и longjmp. Использование set_new_handler(). Объектно-ориентированная обработка исключений. Применение try, catch, throw. Раскрытие стека. Стандартные исключения в C++. Работа с конструкторами и исключениями. Функции terminate(), unexpected().	4
5	2, 3, 4, 7, 14, 15, 16	Раздел 5. Потоки ввода-вывода в C++ Заголовочные файлы. Предопределенные объекты и потоки. Операции помещения и извлечения. Форматирование. Флаги форматирования. Манипуляторы. Ошибки потоков. Файловый ввод-вывод с применением потоков C++. Конструкторы файловых потоков. Открытие файлов в разных режимах. Ввод-вывод в файлы. Форматирование в памяти.	6
6	2, 3, 4, 7, 13, 15	Раздел 6. Шаблоны в C++ Шаблоны функций. Шаблоны классов. Параметры шаблонов. Наследование и шаблоны. Примеры построения шаблонов. Итераторы.	6
7	2, 3, 4, 6, 14, 15, 16	Раздел 7 Библиотека шаблонов STL. Векторы, списки и очереди. Контейнерные классы. Используемые алгоритмы.	6

5.2 Лабораторные работы

№ ЛР	№ раздела	Ссылка на цели	Наименование лабораторных работ	Кол-во часов
1	1	1, 2, 4, 8	Работа с проектами в среде Borland C.	6
2	2	2, 3, 4, 9, 10, 15, 16	Использование классов в C++.	6
3	3	2, 3, 4, 7, 11, 15, 16	Переопределение операций.	6
4	4	2, 3, 4, 7, 12,	Производные классы.	6

		15, 16		
5	5	2, 3, 4, 7, 14, 15, 16	Ввод-вывод в потоки.	6
6	6	2, 3, 4, 7, 13, 15	Обработка исключительных ситуаций на C++.	6
		2, 3, 4, 6, 14, 15, 16		

5.3 Самостоятельное изучение разделов дисциплины

№ раздела	Ссылка на цели Ссылка на цели	Вопросы выносимые на самостоятельное изучение
7	2, 3, 4, 6, 14, 15, 16	Векторы, списки и очереди. Контейнерные классы. Используемые алгоритмы.

6 Учебно-методическое обеспечение дисциплины

6.1 Рекомендуемая литература

6.1.1 Основная литература

1. Страуструп Б. Язык программирования С++. Киев:Диасофт,2001. - 900с.,ил.
2. Бабэ Б. Просто и ясно о Borland С++. Версии 4.0 и 4.5. М.:Бином. - 400с.,ил.
3. Вайнер Р., Пинсон Л. С++ изнутри.-Киев:Диасофт,1993. 304с.,ил.
4. Дьюхарст С. Программирование на С++.-Киев:Диасофт,1993.-272с.,ил.
5. Лукас П. С++ под рукой.-Киев:Диасофт,1993.-176с.,ил.
6. Намиот Д.Е. Основные особенности языка программирования С++. Реализация Turbo С++.- М.:Память,1991.-96с.,ил.
7. Подбельский В.В.Язык С++: Учеб.пособие. - М.: Финансы и статистика, 1995.-560с.:ил.
8. Рассохин Д. От Си к Си++.-М.:Эдель,1993. - 128с.,ил.
9. Романов В.Ю. Программирование на языке С++. Практический подход.- М.:Компьютер,1993. - 160с.,ил.
- 10.Страуструп Б. Язык программирования С++.Часть 1. Киев:Диасофт,1993. - 264с.,ил.
- 11.Страуструп Б. Язык программирования С++.Часть 2. Киев:Диасофт,1993. - 296с.,ил.
- 12.Фейсон Т. Объектно-ориентированное программирование на Borland С+ + 4.5.-К.:Диалектика,1996.-544с., ил.

6.1.2 Дополнительная литература

1. Эллис М., Страуструп Б. Справочное руководство по языку программирования С++ с комментариями. -М.:Мир,1992.-445с.,ил.
2. Bruce Eckel. Thinking in С++.
3. Borland ObjectWindows for С++.Programmer's Guide. version 2.0.

7 Средства обеспечения освоения дисциплины

7.1 Учебно-лабораторное оборудование

Лабораторные работы проводятся в компьютерных классах кафедры ПОВТАС – ауд. №№ 14406 (а, б), 14422, 14423 или 14424.

7.2 Программное обеспечение для выполнения лабораторных работ

- Borland C++ Builder
- Microsoft Visual C++

ЛИСТ
согласования рабочей программы

Направление подготовки: 654600 «Информатика и вычислительная техника»

Специальность: 230105.65 «Программное обеспечение вычислительной техники и автоматизированных систем»

Дисциплина: Объектно-ориентированное программирование на языке C++

Форма обучения: очная

Учебный год 2006 / 2007

Рекомендована заседанием кафедры программного обеспечения вычислительной техники и автоматизированных систем
протокол от « ___ » _____ 2006 г.

Ответственный исполнитель, заведующий кафедрой
ПОВТАС _____ Н.Соловьев « ___ » _____ 2006 г.

Исполнитель
преподаватель кафедры ПОВТАС
_____ Я. Коновал _____ 2006 г.

СОГЛАСОВАНО:

Заведующий кафедрой ПОВТАС _____ Н.Соловьев « ___ » _____ 2006 г.

Председатель методической комиссии по специальности
230105.65 «Программное обеспечение вычислительной техники и
автоматизированных систем»
_____ Н.Соловьев « ___ » _____ 2006 г.

Заведующий отделом комплектования научной библиотеки
_____ « ___ » _____ 2006 г.

Начальник УСИТО
_____ « ___ » _____ 2006 г.