

Министерство образования и науки Республики Казахстан
Карагандинский государственный технический университет

«Утверждаю»
Председатель Ученого совета,
ректор, академик НАН РК
Газалиев А.М.

« ____ » _____ 2015г.

**ПРОГРАММА ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ ДЛЯ СТУДЕНТА
(SYLLABUS)**

по дисциплине **ООР 2205 Объектно-ориентированное программирование**
(код и наименование дисциплины)

модуля **Pr 6 Программирование**
(код и наименование модуля)

для студентов специальности **5B070300 – Информационные системы**
(шифр и наименование специальности)

Факультет информационных технологий

Кафедра информационно-вычислительных систем

Предисловие

Программа обучения по дисциплине для студента (syllabus) разработана:
Ст.преподавателем кафедры ИВС Кайбасовой Д.Ж.

Обсуждена на заседании кафедры ИВС

Протокол № _____ от « _____ » _____ 2015г.

Зав. кафедрой _____ Амиров А.Ж.

(подпись)

Одобрена учебно-методическим советом факультета информационных технологий

Протокол № _____ от « _____ » _____ 2015 г.

Председатель _____ Мустафина Л.М.

Сведения о преподавателе и контактная информация

Кайбасова Динара Женисбековна, старший преподаватель ИВС

(фамилия, имя, отчество преподавателя, ученая степень, ученое звание, должность)

Кафедра ИС находится в главном корпусе КарГТУ (Караганда, б.Мира, 56), аудитория 300а, контактный телефон 56-59-35 (2054), факс -, электронный адрес dindgin@mail.ru

Трудоемкость дисциплины

Форма обучения	Семестр	Количество кредитов	ECTS	Вид занятий					Количество часов СРС	Общее количество часов	Форма контроля
				количество контактных часов			количество часов СРС	всего часов			
				лекции	практические занятия	лабораторные занятия					
Полн	3	3	5	15	15	15	45	90	45	135	РГР
Сокр.	2	3	5	15	15	15	45	90	45	135	РГР

Характеристика дисциплины

Дисциплина «Объектно-ориентированное программирование» входит в модуль специальности рабочего учебного плана государственного общеобразовательного стандарта образования компонент по выбору.

Цель дисциплины

Дисциплина «Объектно-ориентированное программирование» ставит целью формирование понимания идеологии и ключевых аспектов парадигмы объектно-ориентированного программирования (ООП), получение теоретических знаний и практического опыта в технологии объектно-ориентированного программирования на языке программирования высокого уровня C++.

Задачи дисциплины

Задачи дисциплины следующие: формирование у студентов навыков алгоритмического мышления при реализации решения поставленной задачи в виде программы на языке высокого уровня, при этом основной акцент делается на изучении парадигмы объектно-ориентированного программирования (на примере языка C++ и его библиотек) и методов разработки (проектирования) программ.

В результате изучения данной дисциплины студенты должны:

иметь представление:

– об основных концепциях и принципах объектно-ориентированного программирования (ООП);

знать:

– основную терминологию парадигмы ООП,

– эффективные способы разработки программного обеспечения,

- синтаксис и семантику универсального алгоритмического языка программирования высокого уровня,
уметь:
- разрабатывать алгоритмы обработки данных,
- проектировать классы, иерархии классов;
приобрести практические навыки:
- самостоятельного проектирования, кодирования, отладки, тестирования и документирования программ на языке C++.

Пререквизиты

Для изучения данной дисциплины необходимо усвоение следующей дисциплины:

1. Алгоритмы, структуры данных и программирование

Постреквизиты

Знания, полученные при изучении дисциплины «Объектно-ориентированное программирование», используются при освоении следующей дисциплины: «Технология программирования»

Тематический план дисциплины

Наименование раздела, (темы)	Трудоемкость по видам занятий, ч.				
	лекции	Практические	Лабораторные	СРСП	СРС
1 Вводная лекция. Предмет курса, история, текущее состояние и перспективы ООП	1			3	1
2 Объектно-ориентированное программирование, основные концепции: абстрагирование, инкапсуляция, модульность, иерархия, типизация, параллелизм, сохраняемость.	1			3	3
3 Объекты. Состояние, поведение, идентичность, жизненный цикл объекта. Отношения между объектами.	1			3	3
4 Классы. Структура класса Абстрактные классы, интерфейсы и классы-утилиты. Отношения между классами: ассоциация, агрегация и композиция, использование, наследование.	1			3	3
5 Описание класса Описание объектов. Члены класса, управление доступом, конструкторы, деструктор. Встроенные функции, копирование объектов, объекты-члены, эффективные пользовательские типы.	2			3	2
6 Перегрузка операторов. Операторные функции, бинарные и унарные операторы, операторы-члены и не-	2			3	3

члены, предопределенный смысл операторов.					
7 Производные классы. Наследование. Иерархия классов. Виртуальные функции, абстрактные классы. Управление доступом при наследовании, правила доступа, множественное наследование.	2			3	3
8 Шаблоны классов. (Базовые механизмы, Инстанцирование, Параметры шаблонов, Специализация шаблонов классов, Минимизация кода, Наследование и шаблоны).	2			3	2
9 Обработка ошибок. Исключения. (throw и catch, Выбор исключений, Группировка исключений, Перехват исключений, Повторная генерация, Исключения в конструкторах, Исключения в деструкторах).	2			3	2
10 Преобразование типов. (dynamic_cast, static_cast, reinterpret_cast, const_cast)	1			3	3
11 Классы, данные, методы классов, доступ к методам и классам.					
12 Конструкторы и деструкторы классов		2		3	
13 Наследование. Простое наследование, защита и сокрытие данных в производных классах, конструкторы производных классов.		2		3	
14 Полиморфизм. Перегрузка операторов, дружественные классы		2		3	
15 Объектно-ориентированные средства разработки программ		2		3	
16 Простые приложения. Главное окно приложения		2		3	
17 Параметризованные типы		2			
18 Шаблоны функций, шаблоны классов		3			
19 Простейшие классы и объекты			3		4
20 Классы. Перегрузка операций.			3		4
21 Иерархии классов, наследование.			3		4
22 Шаблоны. Реализация шаблонов. Обработка исключительных ситуаций			3		4
23 Файловые и строковые потоки			3		4
ИТОГО:	15	15	15	45	45

Перечень практических занятий

1. Конструкторы и деструкторы классов
2. Наследование. Простое наследование, защита и сокрытие данных в производных классах, конструкторы производных классов.

3. Полиморфизм. Перегрузка операторов, дружественные классы
4. Объектно-ориентированные средства разработки программ
5. Простые приложения. Главное окно приложения
6. Параметризованные типы
7. Шаблоны функций, шаблоны классов

Перечень лабораторных занятий

- 1 Простейшие классы и объекты
- 2 Классы. Перегрузка операций
- 3 Иерархии классов, наследование
- 4 Шаблоны. Реализация шаблонов Обработка исключительных ситуаций
- 5 Файловые и строковые потоки

Тематический план самостоятельной работы студента с преподавателем

Наименование темы СРСП	Цель занятия	Форма проведения занятия	Содержание задания	Рекомендуемая литература
1 Текущее состояние и перспективы ООП	Получение практических навыков	Выполнение индивидуальных заданий	Анализ и постановка задач в терминах ООП	[1,2]
2 Объектно-ориентированное про-граммирование, основные концепции: абстрагирование, инкапсуляция, модульность, иерархия, типизация, параллелизм, сохраняемость.	Получение практических навыков	Выполнение индивидуальных заданий	Разработка классов в соответствии с основными принципами парадигмы ООП	[1, 2]
3 Объекты. Состояние, поведение, идентичность, жизненный цикл объекта. Отношения между объектами.	Получение практических навыков	Выполнение индивидуальных заданий	Разработка пользовательских классов, конструкторов и деструкторов	[1, 2, 4]
4 Классы. Структура класса Абстрактные классы, интерфейсы и классы-утилиты. Отношения между классами: ассоциация, агрегация и композиция, использование, наследование.	Получение практических навыков	Выполнение индивидуальных заданий	Определение состава классов для решения задач, их типов, построение иерархии.	[4,5]
5 Описание класса Описание объектов. Члены класса, управление доступом, конструкторы, деструктор. Встроенные функции, копирование объектов, объекты-члены, эффективные пользовательские типы.	Получение практических навыков	Выполнение индивидуальных заданий	Проектирование классов в соответствии с требованием инкапсуляции, реализация интерфейсов и функций.	[4,5]

6 Перегрузка операторов. Операторные функции, бинарные и унарные операторы, операторы-члены и не-члены, предопределенный смысл операторов.	Получение практических навыков	Выполнение индивидуальных заданий	Разработка классов с предопределенным смыслом операторов.	[2,4,5]
7 Производные классы. Наследование. Иерархия классов. Виртуальные функции, абстрактные классы. Управление доступом при наследовании, правила доступа, множественное наследование.	Получение практических навыков	Выполнение индивидуальных заданий	Изучение механизмов наследования, реализация полиморфизма.	[2,4,5]
8 Шаблоны классов. (Базовые механизмы, Инстанцирование, Параметры шаблонов, Специализация шаблонов классов, Минимизация кода, Наследование и шаблоны).	Получение практических навыков	Выполнение индивидуальных заданий	Разработка шаблонов классов и функций.	[2,4,5]
9 Обработка ошибок. Исключения. (throw и catch, Выбор исключений, Группировка исключений, Перехват исключений, Повторная генерация, Исключения в конструкторах, Исключения в деструкторах).	Получение практических навыков	Выполнение индивидуальных заданий	Изучение методов разработки надежного программного средства.	[2,4,5]
10 Преобразование типов. (dynamic_cast, static_cast, reinterpret_cast, const_cast)	Получение практических навыков	Выполнение индивидуальных заданий	Изучение механизмов явного приведения типов.	[2,4,5]

Темы контрольных заданий для СРС

1. Разработать классы для заданных объектов в соответствии с заданием
2. Определить и реализовать класс - абстрактный тип данных.
3. Дополнить определение класса заданными перегруженными операциями (в соответствии с вариантом).
4. Создать иерархию классов с использованием статических компонентов класса.
5. Для определения иерархии классов связать отношением наследования классы, приведенные в приложении (для заданного варианта).
6. Разработать иерархию классов в виде графа
7. Создать шаблон заданного класса и использовать его для данных различных типов.

Критерии оценки знаний студентов

Экзаменационная оценка по дисциплине определяется как сумма максимальных показателей успеваемости по рубежным контролям (60%) и итоговой аттестации (экзамен) (40%) и составляет значение 100% в соответствии с таблицей.

График выполнения и сдачи заданий по дисциплине

Вид контроля	Цель и содержание задания	Рекомендуемая литература	Продолжительность выполнения	Форма контроля	Срок сдачи	Баллы
Посещаемость лекций	Усвоение материала по темам лекций		15 контактных часов	Текущий	На каждой лекции	
Сдача лабораторных работ №№ 1-5	Усвоение материала по дисциплине	МУ к выполнению лабораторных работ	15 контактных часов	Текущий	3, 6, 9, 12,14 недели	25
Сдача практических работ	Усвоение материала по дисциплине	МУ к выполнению практических работ	15 контактных часов	Текущий	2, 4, 6, 8,10, 12, 14 недели	10
Задания к темам СРСП	Получение практических навыков по языку С++ и ООП	Согласно тематики СРСП	45 контактных часов	Текущий	Еженедельно	15
Теоретический модуль	Проверка усвоения материала дисциплины	Конспект лекций	0,5 контактных часа	Рубежный	7,14 неделя	20
РГР	Проверка усвоения материала дисциплины	Весь перечень основной и дополнительной литературы	2 контактных часа	Итоговый	15 неделя	40
ИТОГО						100

Политика и процедуры

При изучении дисциплины «Объектно-ориентированное программирование» прошу соблюдать следующие правила:

1 Не опаздывать на занятия.

2 Не пропускать занятия без уважительной причины, в случае болезни прошу представить справку, в других случаях – объяснительную записку.

3 В обязанности студента входит посещение всех видов занятий.

4 Согласно календарному графику учебного процесса сдавать все виды контроля.

5 Пропущенные практические и лабораторные занятия отрабатывать в указанное преподавателем время.

Список основной литературы

1 Павловская Т. А. С/С++. Процедурное и объектно-ориентированное программирование: учеб. для студ. вузов / Т. А. Павловская. -СПб.: Питер, 2015.

2. С++. Объектно-ориентированное программирование : учеб. пособие для студ. вузов / В. В. Лаптев. -СПб.: Питер, 2008.

3. Основные концепции и механизмы объектно-ориентированного программирования: Теория и технология программирования: учеб. пособие / Е. В. Пышкин. - СПб. : БХВ - Петербург, 2005.

4. Объектно-ориентированное программирование с использованием с++, пер. англ. / И. Пол. - КИЕВ : НИПФ «ДИАСОФТ ЛТД», 1995

5. Б. Страуструп Язык программирования С++, 3-е изд./Пер. с англ. — Специальное изд. - М. : БИНОМ, 2006

6. Б. Страуструп Дизайн и эволюция языка С++, Пер. с англ. - ДМК Пресс, Питер, 2006.

7. Скотт Мейерс, Эффективное использование С++. 50 рекомендаций по улучшению ваших программ и проектов, Пер. с англ. - ДМК, 2006

8. Скотт Мейерс, Эффективное использование С++. 35 новых способов улучшить стиль программирования, Пер. с англ. - ДМК, 2006

Список дополнительной литературы

9 С/С++. Структурное и объектно-ориентированное программирование : практикум / Т. А. Павловская, Ю. А. Щупак. - М. ; СПб. ; Нижний Новгород : Питер, 2011.

10 Практикум по объектно-ориентированному программированию: учебное пособие / И. А. Бабушкина, С. М. Окулов. - М. : Бином. Лаборатория знаний, 2004.

11 Приемы объектно-ориентированного проектирования. Паттерны проектирования: научное издание: пер. с англ. / Э. Гамма [и др.]. - М. ; СПб. ; Нижний Новгород : Питер, 2007.

12 С++. Объектно-ориентированное программирование: учебное пособие / В. В. Лаптев. - М. ; СПб. ; Нижний Новгород : Питер, 2008

**ПРОГРАММА ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ ДЛЯ СТУДЕНТА
(SYLLABUS)**

по дисциплине **ООР 2205 Объектно-ориентированное программирование**
(код и наименование дисциплины)

модуля **Pr 6 Программирование**
(код и наименование модуля)

Гос.изд.лиц. № 50 от 31.03.2004. Подписано в печать _____.____.15г. Формат 60x90/16
Усл.печ.л. 0,94 Тираж Цена договорная

Издательство Карагандинского государственного технического университета
100027, Караганда, б.Мира, 56