

Правительство Российской Федерации

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

Факультет Прикладной математики и кибернетики

Программа дисциплины Объектно-ориентированное программирование

для направления 230700.62 "Прикладная информатика" подготовки бакалавра

Автор программы:

Аксенов С. А., к.т.н., saksenov@hse.ru

Одобрена на заседании кафедры Механики и математического моделирования

«__»_____ 20 г

Зав. кафедрой Е.Н. Чумаченко

Рекомендована секцией УМС [Введите название секции УМС] «__»_____ 20 г

Председатель [Введите И.О. Фамилия]

Утверждена УС факультета Прикладной математики и кибернетики «__»_____ 20 г.

Ученый секретарь [Введите И.О. Фамилия] _____ [подпись]

Москва, 2012

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



1 Область применения и нормативные ссылки

Настоящая программа учебной дисциплины устанавливает минимальные требования к знаниям и умениям студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих данную дисциплину, учебных ассистентов и студентов направления подготовки 230700.62 "Прикладная информатика", изучающих дисциплину Объектно-ориентированное программирование.

Программа разработана в соответствии с:

- ФГОС по направлению подготовки бакалавра 230700.62 "Прикладная информатика", уровень подготовки - бакалавр;
- Образовательной программой 230700.62 "Прикладная информатика" подготовки бакалавра.
- Рабочим учебным планом университета по направлению 230700.62 "Прикладная информатика" подготовки бакалавра, утвержденным в 2013 г.

2 Цели освоения дисциплины

Целями освоения дисциплины Объектно-ориентированное программирование являются

- Формирование у слушателей дисциплины представления о возможностях и особенностях языков программирования при проектировании, разработке и отладке компьютерных программ;
- Изучение возможностей и особенностей языков программирования при разработке и реализации алгоритмов, создании и отладке компьютерных программ;
- Формирование представления о принципах применения информационных технологий, преимуществах и недостатках различных языков программирования при решении прикладных задач.

3 Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины студент должен:

- **Знать** парадигмы и методологии программирования, особенности объектно-ориентированных языков программирования, наиболее широко используемых средств программирования; концепции, синтаксической и семантической организации, методы использования современных объектно-ориентированных языков программирования; теоретические и методические основы, понимать функциональные возможности в области разработки информационных систем;
- **Уметь** применять в профессиональной деятельности современные объектно-ориентированные языки программирования; профессионально решать задачи производственной и технологической деятельности с учетом современных достижений науки и техники, включая: разработку алгоритмических и программных решений в области системного и прикладного программирования; разработку математических, информационных и имитационных моделей по тематике выполняемых исследований;
- **Владеть** базовыми математическими знаниями и информационными технологиями, эффективно применять их для решения научно-технических задач и прикладных задач, связанных с развитием и использованием информационных технологий;

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
Способен ставить и решать прикладные задачи с использованием современных информационно-коммуникационных технологий	ПК-4	Применяет современные информационные технологии при постановке и решении прикладных задач	Лекции, практические занятия, самостоятельная работа
Способен моделировать и проектировать структуры данных и знаний, прикладные и информационные процессы	ПК-9	Применяет навыки проектирования структур данных при разработке программных продуктов	Лекции, практические занятия, самостоятельная работа
Способен применять к решению прикладных задач базовые алгоритмы обработки информации, выполнять оценку сложности алгоритмов, программировать и тестировать программы	ПК-10	Применяет базовые математические знания и алгоритмы при создании, отладке и тестировании программных продуктов	Лекции, практические занятия, самостоятельная работа
Способен принимать участие в создании и управлении ИС на всех этапах жизненного цикла	ПК-11	Демонстрирует понимание особенностей жизненного цикла разработки информационных систем	Практические занятия, самостоятельная работа
Способен работать с информацией в глобальных компьютерных сетях	ОК-8	Применяет поиск в сети Интернет и других источниках при необходимости получения информации о новейших инструментах и библиотеках, обеспечивающих необходимый для решения поставленной задачи функционал	Практические занятия, самостоятельная работа

4 Место дисциплины в структуре образовательной программы

Дисциплина относится к профессиональному учебному циклу (Б.3) и блоку дисциплин, обеспечивающих базовую (общепрофессиональную) подготовку.

Изучение данной дисциплины базируется на следующих дисциплинах:

- Языки и методы программирования
- Практикум на ЭВМ

Для освоения учебной дисциплины, студенты должны владеть следующими знаниями и компетенциями:

- Владеть основами программирования на языке С.

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



- Программная инженерия
- Проектирование информационных систем.
- Программирование для Интернет.
- Преддипломная практика.

5 Тематический план учебной дисциплины

№	Название раздела	Всего часов	Аудиторные часы			Самостоятельная работа
			Лекции	Семинары	Практические занятия	
1.	Введение в программирование на C++.	11	2		4	5
2.	Знакомство с библиотекой STL.	16	3		6	7
3.	Построение пользовательских типов данных.	22	4		8	10
4.	Наследование, виртуальные функции, полиморфизм.	27	5		10	12
5.	Шаблоны C++.	11	2		4	5
6.	Отладка программ в Visual Studio. Обработка исключений.	11	2		4	5
7.	Концепция программирования для Windows. Знакомство с библиотекой MFC.	11	2		4	5
8.	Архитектура документ-представление.	17	3		6	8
9.	Элементы пользовательского интерфейса.	22	4		8	10
10.	Основы языка Java.	11	2		4	5
11.	Особенности ООП на Java.	17	3		6	8
12.	Разработка пользовательского интерфейса Java.	22	4		8	10



Формы контроля знаний студентов

Тип контроля	Форма контроля	1 год				Параметры **
		1	2	3	4	
Текущий (неделя)	Домашнее задание	4	4	4	4	Создание программы в IDE и защита её текста, 80 минут
Промежуточный	Зачет	*				Создание программы в IDE и защита её текста, 80 минут
Итоговый	Экзамен				*	Создание программы в IDE и защита её текста, 80 минут

5.1 Критерии оценки знаний, навыков

Домашнее задание: Студент должен продемонстрировать способность разработать программу по заранее выданному заданию преподавателя и защитить её при опросе.

Зачет: Студент должен продемонстрировать умение создавать компьютерные программы на языке C++ основе информации по темам 1-3. Компетенции: ПК-4, ПК-9, ПК-10, ПК-11, ОК-8.

Экзамен: Студент должен продемонстрировать умение создавать компьютерные программы и знание принципов ООП в рамках тем 1-12. Компетенции: ПК-4, ПК-9, ПК-10, ПК-11, ОК-8.

Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

Проведение контроля должно осуществляться в компьютерном классе с установленным IDE.

5.2 Порядок формирования оценок по дисциплине

(подробные методические рекомендации по формированию оценок по дисциплине приведены в приложении)

Преподаватель оценивает работу студентов на семинарах и практических занятиях: оценивается активность студента в дискуссиях, скорость и правильность решения задач. Оценки за работу на семинарских и практических занятиях преподаватель выставляет в рабочую ведомость. Накопленная оценка по 10-ти балльной шкале за работу на семинарских и практических занятиях определяется перед промежуточным или итоговым контролем - $O_{аудиторная}$.

Преподаватель оценивает самостоятельную работу студентов: оценивается полнота и правильность выполнения домашних работ, уровень ориентированности студента в демонстрируемых им программах. Оценки за самостоятельную работу студента преподаватель выставляет в рабочую ведомость. Накопленная оценка по 10-ти балльной шкале за самостоятельную работу определяется перед промежуточным или итоговым контролем - $O_{сам. работа}$.

Накопленная оценка за текущий контроль учитывает результаты студента по текущему контролю следующим образом:

$$O_{накопленная} = 0.6 * O_{текущий} + 0.4 * O_{ауд}$$

где $O_{текущий}$ рассчитывается как взвешенная сумма всех форм текущего контроля, предусмотренных в РУП:



$$O_{\text{текущий}} = O_{\text{д/з}} ;$$

Результирующая оценка за дисциплину рассчитывается следующим образом:

$$O_{\text{промежуточная } i} = 0.5 \cdot O_{\text{накопленная } i \text{ этапа}} + 0.5 \cdot O_{\text{промежуточный зачет/экзамен}}$$

Где $O_{\text{накопленная } i \text{ этапа}}$ рассчитывается по приведенной выше формуле

$$O_{\text{накопленная Итоговая}} = (O_{\text{промежуточная } 1} + O_{\text{промежуточная } 2}) / 2$$

Способ округления накопленной оценки промежуточного (итогового) контроля в форме экзамена: арифметический.

На передаче студенту не предоставляется возможность получить дополнительный балл для компенсации оценки за текущий контроль.

На зачете студент может получить дополнительный вопрос (дополнительную практическую задачу, решить к передаче домашнее задание), ответ на который оценивается в 1 балл.

На экзамене студент может получить дополнительный вопрос (дополнительную практическую задачу, решить к передаче домашнее задание), ответ на который оценивается в 1 балл.

В диплом выставляет результирующая оценка по учебной дисциплине, которая формируется по следующей формуле:

$$O_{\text{результ}} = 0.5 \cdot O_{\text{накопл}} + 0.5 \cdot O_{\text{итоговый}}$$

Способ округления результирующей оценки по учебной дисциплине: арифметический.

ВНИМАНИЕ: оценка за итоговый контроль **блокирующая**, при неудовлетворительной итоговой оценке она равна результирующей.

6 Содержание дисциплины

Раздел представляется в удобной форме (список, таблица). Изложение строится по разделам и темам. Содержание темы может распределяться по лекционным и практическим занятиям.

№	Наименование раздела дисциплины	Содержание раздела	Аудиторная работа	Самостоятельная работа	Литература к разделу
1.	Введение в программирование на C++.	Пространства имен. Ввод/вывод в C++. Операции инкремента и декремента. Особенности операторов инкремента и декремента при работе с указателями. Ссылки. Выделение/освобождение памяти в C++.	6	5	[1,2]



№	Наименование раздела дисциплины	Содержание раздела	Аудиторная работа	Самостоятельная работа	Литература к разделу
2.	Знакомство с библиотекой STL.	Понятие объекта и класса. Классы стандартной библиотеки для работы со строками. Классы стандартной библиотеки для работы с контейнерами. Операции с объектами стандартной библиотеки. Преобразование типов.	9	7	[1,2]
3.	Построение пользовательских типов данных.	Создание пользовательских классов. Принцип композиции при конструировании новых классов. Инициализация данных объекта класса (Конструктор по умолчанию, конструктор с параметрами). Назначение деструктора. Способы передачи объекта в функцию, конструктор копирования. Перегрузка операторов. Статические переменные и методы класса. Константные методы. Перегруженные методы.	12	10	[1,4]
4.	Наследование, виртуальные функции, полиморфизм.	Спецификаторы доступа public, protected, private. Понятие и механизм наследования классов. Преимущества и недостатки наследования по сравнению с композицией. Виртуальные функции, понятие полиморфизма. Чистые виртуальные функции и абстрактные классы. Доступ к объекту через указатель, преобразование типов. Непрямые базовые классы. Виртуальный деструктор. Множественное наследование.	15	12	[1,2]
5.	Шаблоны C++.	Понятие обобщенного программирования. Определение шаблона класса. Создание объектов шаблона класса. Наследование шаблонных классов. Шаблоны функций.	6	5	[1,2]
6.	Отладка программ в Visual Studio. Обработка исключений.	Трассировка программы в Visual Studio. Макросы ASSERT, VERIFY и TRACE. Понятие исключения. Механизм генерации и обработки исключений.	6	5	[1,2]
7.	Концепция программирования для Windows. Знакомство с библиотекой MFC.	Функция WinMain. Цикл обработки сообщений. Минимальная программа для Windows. Использование библиотеки MFC. Класс CApp. Основные элементы программ на базе MFC. Типы классов окон MFC. Основные сообщения Windows. Графические возможности GDI	6	5	[1]
8.	Архитектура документ-представление.	Использование мастера Visual Studio для создания SDI и MDI приложений. Архитектура SDI приложений. Архитектура MDI приложений. Различные представления одного документа. Уведомление представлений об изменении документа. Понятие сериализации, сохранение и печать документа.	9	8	[1]



№	Наименование раздела дисциплины	Содержание раздела	Аудиторная работа	Самостоятельная работа	Литература к разделу
9.	Элементы пользовательского интерфейса.	Создание и работа с меню. Создание и работа с панелями инструментов. Поддерживаемые элементы управления (кнопки, текстовые поля, списки и т.д.). Создание, редактирование, получение уведомлений от элементов управления.	12	10	[1]
10.	Основы языка Java.	Концепция технологии Java. Типы переменных в Java. Жизненный цикл переменной. Условные переходы и циклы. Выделение памяти. Массивы и строки. Организация кода Java, пакеты и подпакеты.	6	5	[3]
11.	Особенности ООП на Java.	Инициализация данных класса. Блоки инициализации и конструкторы. Конструкторы базовых классов. Наследование в Java. Абстрактные классы. Интерфейсы.	9	8	[3]
12.	Разработка пользовательского интерфейса Java.	«Легкий» и «тяжелый» пользовательские интерфейсы. Иерархия классов пользовательского интерфейса библиотеки awt, понятие компонента и контейнера. Менеджеры размещения. Инструменты для работы с графикой. Обработка событий в Java.	12	10	[3]

Занятия проводятся в компьютерном классе в интерактивной форме, включают в себя дискуссии по текущим заданиям и решение задач на ЭВМ.

7 Образовательные технологии

Лекции должны проводиться в классах, обеспеченных компьютером и проекционным оборудованием.

Практические занятия должны проводиться в компьютерных классах оснащенных необходимым программным обеспечением (средствами разработки программ) и включать в себя дискуссии по текущим заданиям и решение задач на ЭВМ.

7.1 Методические рекомендации преподавателю

Не предусмотрены.

7.2 Методические указания студентам

Не предусмотрены.

8 Оценочные средства для текущего контроля и аттестации студента

8.1 Вопросы для оценки качества освоения дисциплины

Примерный перечень вопросов к зачету (экзамену) по всему курсу или к каждому промежуточному и итоговому контролю для самопроверки студентов.

1. Пространства имен. Ввод/вывод в C++.
2. Операции инкремента и декремента. Особенности операторов инкремента и декремента при работе с указателями.



3. Выделение/освобождение памяти в C++.
4. Классы стандартной библиотеки для работы со строками.
5. Классы стандартной библиотеки для работы с контейнерами.
6. Преобразование типов в C++/
7. Принципы наследования и композиции при конструировании новых классов. Преимущества и недостатки наследования по сравнению с композицией.
8. Инициализация данных объекта класса (Конструктор по умолчанию, конструктор с параметрами). Назначение деструктора. Способы передачи объекта в функцию, конструктор копирования.
9. Перегрузка операторов.
10. Статические переменные и методы класса. Константные методы. Перегруженные методы.
11. Виртуальные функции, понятие полиморфизма. Чистые виртуальные функции и абстрактные классы. Виртуальный деструктор.
12. Множественное наследование.
13. Понятие обобщенного программирования. Шаблоны классов. Шаблоны функций.
14. Макросы ASSERT, VERIFY и TRACE. Понятие исключения. Механизм генерации и обработки исключений.
15. Функция WinMain. Цикл обработки сообщений. Минимальная программа для Windows.
16. Архитектура приложений MFC. Класс CApp. Классы CWindow и CFrame.
17. Основные сообщения Windows. Механизм обработки сообщений в MFC.
18. Рисование в окне, инструменты GDI.
19. Основные классы архитектуры SDI и принцип их взаимодействия.
20. Основные классы архитектуры MDI и принципы их взаимодействия.
21. Понятие сериализации, сохранение и печать документа.
22. Создание и работа с меню. Использование ресурсов проекта.
23. Создание и работа с панелями инструментов. Использование ресурсов проекта.
24. Основные принципы работы с элементами управления (создание, размещение, получение уведомлений, изменение).
25. Концепция технологии Java. Типы переменных в Java.
26. Массивы и строки Java. Организация кода Java, пакеты и подпакеты.
27. Наследование в Java. Абстрактные классы. Интерфейсы.
28. «Легкий» и «тяжелый» пользовательские интерфейсы. Иерархия классов пользовательского интерфейса библиотеки awt, понятие компонента и контейнера. Менеджеры размещения.
29. Механизм обработки событий в Java.

9 Учебно-методическое и информационное обеспечение дисциплины

9.1 Основная литература

1. Ivor Horton, Beginning Visual C++ 2010, Indianapolis : Wiley Publishing, Inc., 2011,
2. P.J. Deitel, H.M. Deitel, C++ How to Program, 7th Pearson Education, 2010,
3. Bruce Eckel, Thinking in Java. Prentice Hall PTR, 2009.

9.2 Дополнительная литература

4. Э. Кенинг, Б. Му, Эффективное программирование на C++ // М. : «Вильямс», 2002,
5. Айвор Хортон, Visual C++ 2010 полный курс, Москва : Вильямс, 2011
6. С. Мейерс, Эффективное использование C++: 50 рекомендаций по улучшению ваших программ и проектов // Москва : Питер, 2006,



7. С. Мейерс, Эффективное использование STL // Москва : Питер, 2002,
8. Г. Саттер, Решение сложных задач на С++ // Москва : Вильямс, 2002,
9. А. Александреску, Современное проектирование на С++ // Москва : Вильямс, 2002,
10. Deitel P. J. , Deitel H. M., Java How to Program, 9th Edition, Prentice Hall, 2011,
11. Дейтел Х.М., Дейтел П.Дж., Сантри С.И. Технологии программирования на Java. Том 1. Графика, JavaBeans, интерфейс пользователя // М. : «Бином-пресс», 2003,
12. Дейтел Х.М., Дейтел П.Дж., Сантри С.И. Технологии программирования на Java. Том 2. Распределенные приложения // М. : «Бином-пресс», 2003,
13. Дейтел Х.М., Дейтел П.Дж., Сантри С.И. Технологии программирования на Java. Том 3. Корпоративные системы // М. : «Бином-пресс», 2003,

10 Материально-техническое обеспечение дисциплины

Для проведения лекций необходима аудитория, оснащенная компьютером и проекционным оборудованием.

Для проведения практических занятий необходим компьютерный класс, оснащенный современным программным обеспечением, IDE для разработки программ на Java, Visual C++ 2010.

Методические рекомендации по формированию оценок по дисциплине

Данные методические рекомендации составлены на основании Положения об организации контроля знаний, утвержденного УС НИУ ВШЭ от 24.06.2011, протокол №26.

1) Структура оценки по дисциплине согласно положению об организации контроля знаний:





2) Таблица 1. Формирование оценки по дисциплине: если дисциплина читается 1 этап (модуль)

Элемент оценки	Накопленная оценка			Итоговая оценка за экзамен/зачет	Результирующая оценка за дисциплину (Выставляется в дипломе)		
	Текущий контроль	Аудиторная работа (Лекции, практические занятия, семинарские занятия)	Самостоятельная внеаудиторная работа студентов				
Действия преподавателя	1	Выставление оценки в 10-балльной системе по каждой форме текущего контроля (эссе, контрольная работа, домашнее задание, реферат, коллоквиум)	Выставление оценки $O_{\text{ауд}}$ по 10-балльной шкале за аудиторную работу студента. ВАЖНО: в НИУ ВШЭ в рамках аудиторной работы не оценивается посещение лекций, семинарских занятий и практических занятий, а только работа студента. (Оценка выставляется только при решении преподавателя оценивать данный вид деятельности студента)	Выставление оценки $O_{\text{сам.работа}}$ по 10-балльной шкале за аудиторную работу студента. (Оценка выставляется только при решении преподавателя оценивать данный вид деятельности студента)	Выставление оценки за итоговый контроль (зачет/экзамен) в 10 балльной системе	1	Определение весов q_1 и q_2 (ВНИМАНИЕ, Сумма удельных весов должна быть равна единице: $\sum q_i = 1$, при этом, $0,2 \leq q_i \leq 0,8$)
	2	Определение весов n_i (ВНИМАНИЕ, сумма $n_i = 1$)				2	$O_{\text{результ}} = q_1 \cdot O_{\text{итог.контроль}} + q_2 \cdot O_{\text{накопленная}}$
	3	Расчет оценки за текущий контроль $O_{\text{текущий}} = n_1 \cdot O_{\text{эссе}} + n_2 \cdot O_{\text{к/р}} + n_3 \cdot O_{\text{реф}} + n_4 \cdot O_{\text{кол}} + n_5 \cdot O_{\text{дз}}$					
	Определение весов k_1 k_2 k_3 (ВНИМАНИЕ, сумма $k_i = 1$, в случае, если преподаватель не учитывает аудиторную и самостоятельную внеаудиторную работу студентов, то k_2 и k_3 равны 0 (нулю), а $k_1 = 1$).						
	Расчет накопленной оценки $O_{\text{накопленная}} = k_1 \cdot O_{\text{текущий}} + k_2 \cdot O_{\text{ауд}} + k_3 \cdot O_{\text{сам.работа}}$						
Что получается в результате	$O_{\text{накопленная}}^*$			$O_{\text{итог.контроль}}$	$O_{\text{результирующая}}^*$		

3) Формирование оценки по дисциплине, если она читается несколько этапов (модулей) поясним на примере дисциплины читаемой 3 этапа (таблица 2).



Таблица 2. Формирование оценки по дисциплине: если дисциплина читается несколько этапов (модулей)

Элемент оценки		Промежуточная оценка за 1 этап			Промежуточная оценка за 2 этап			Накопленная оценка 3 (за 3 тап)			Итоговая оценка за экзамен/зачет	Результирующая оценка за дисциплину (Выставляется в диплом)
		Текущий контроль	Аудиторная работа	Самостоятельная вне-аудиторная работа студентов	Оценка за экзамен/зачет (по окончанию этапа 1) (ВАЖНО! Не является блокирующей)	Текущий контроль	Аудиторная работа	Самостоятельная вне-аудиторная работа студентов	Оценка за экзамен/зачет (по окончанию этапа 2) (ВАЖНО! Не является блокирующей)	Текущий контроль		
Действия преподавателя		действия преподавателя в рамках каждого этапа соответствуют действию преподавателя по формированию оценки, если дисциплина читается один этап (модуль) (таблица 1)			действия преподавателя в рамках каждого этапа соответствуют действию преподавателя по формированию оценки, если дисциплина читается один этап (модуль) (таблица 1)			действия преподавателя (таблица 1)			Выставление оценки за итоговый контроль (зачет/экзамен) в 10 балльной системе	Определение весов q_1 и q_2 (ВНИМАНИЕ, Сумма удельных весов должна быть равна единице: $\sum q_i = 1$, при этом, $0,2 \leq q_i \leq 0,8$)
Результат		$O_{\text{промежуточная 1}}^*$			$O_{\text{промежуточная 2}}^*$			$O_{\text{накопленная 3}}^*$				$O_{\text{итог. контроль}}$
ИТОГ		$O_{\text{накопленная Итоговая}} = (O_{\text{промежут 1}} + O_{\text{промежут 2}} + O_{\text{накопленная 3}}) : \text{кол-во модулей}$ Среднее арифметическое от суммы оценок.										

* способ округления оценки должен быть указан в программе учебной дисциплины