

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Воронежский государственный технический университет»

УТВЕРЖДАЮ

Декан факультета экономики менеджмента и
информационных технологий

С.А.Баркалов

«30» августа 2017 г.

РАБОЧАЯ ПРОГРАММА

дисциплины

«Среды объектно-ориентированного программирования»

Направление подготовки 09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И
ТЕХНОЛОГИИ

Профиль Информационные системы и технологии строительстве

Квалификация выпускника Бакалавр

Нормативный период обучения 4 года

Форма обучения очная

Год начала подготовки 2017

Автор программы

 /Смолянинов А.В./

Заведующий кафедрой
Информационных технологий
и автоматизированного
проектирования в
строительстве

 /Смолянинов А.В./

Руководитель ОПОП

 /Курипта О.В./

Воронеж 2017

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Целью дисциплины является приобретение базовых знаний и навыков программирования, проектирование и разработка приложений с применением объектно-ориентированного подхода, приобретение навыков, позволяющих будущим специалистам вести успешную разработку специализированного программного обеспечения в тех областях и сферах деятельности, в которых они будут трудиться.

Изучение дисциплины должно способствовать формированию у студентов основ научного мышления, в том числе: владение основными методами, способами и инструментами создания программного обеспечения, использования для решения практических задач.

1.2. Задачами освоения дисциплины являются:

- ознакомление с тенденцией развития программного обеспечения;
- обучение разработке алгоритмов на основе объектно-ориентированного подхода в визуальных средах разработки;
- освоение языка программирования высокого уровня, а также стандартов кодирования, спецификаций и последующих решений на их основе;
- работа с современными программными инструментальными средствами разработки;
- изучение особенностей разработки программного обеспечения под имеющиеся операционные системы, в том числе и мобильные платформы.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Среды объектно-ориентированного программирования» относится к дисциплинам базовой части блока Б1.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Среды объектно-ориентированного программирования» направлен на формирование следующих компетенций:

ОПК-1 - владением широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий

ОПК-6 - способностью выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно) для решения поставленной задачи

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-1	знать алгоритмы, методики подготовки к работе информационных систем, их типы
	уметь поддерживать работоспособность

ОПК-6	информационных систем
	владеть приемами тестирования технических средств
	знать основные характеристики аппаратных средств ЭВМ информационных систем
	уметь осуществлять их настройку и выполнять конфигурирование указанных систем
	владеть приемами тестирования технических средств

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Среды объектно-ориентированного программирования» составляет 5 з.е.

Распределение трудоемкости дисциплины по видам занятий
очная форма обучения

Виды учебной работы	Всего часов	Семестры
		2
Аудиторные занятия (всего)	54	54
В том числе:		
Лекции	18	18
Лабораторные работы (ЛР)	36	36
Самостоятельная работа	90	90
Курсовой проект	+	+
Часы на контроль	36	36
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость:		
академические часы	180	180
зач.ед.	5	5

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий
очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Интегрированные среды и средства разработки на языке программирования высокого уровня	Интерфейс и конфигурация интегрированной среды разработки (IDE) под решение вычислительных задач. Разработка консольных проектов и приложений с графическим интерфейсом.	4	6	14	24
2	Базовые конструкции и ключевые слова языка программирования высокого уровня	Типы данных языка и расширения типов, предоставляемые средой. Арифметические операции. Правила именования методов, переменных и констант. Использование структур, условных конструкций, циклов, функций, а также директив предпроцессора.	4	6	14	24
3	Объектно-ориентированный подход к разработке программных средств	Описание и свойства объектов. Конструкторы, деструкторы классов, объявления статических объектов, переменных и методов в классе. Наследование, виды наследования. Абстрактные классы. Множественное наследование.	4	6	14	24
4	Разработка приложений с графическими интерфейсными	Основы разработки программных интерфейсов, средства разработки, иерархия и структура интерфейсных классов. Средства взаимодействия	2	6	16	24

	возможностями	между приложением и действиями, исходящими от пользователя				
5	Создание подсистем на основе подключаемых библиотек	Динамически-подключаемые источники данных. Работа с системными библиотеками операционных систем	2	6	16	24
6	Построение кроссплатформенных приложений	Технологии и инструменты сред разработки по созданию кроссплатформенных приложений. Использование базовых классов среды. Перенос данных.	2	6	16	24
Итого			18	36	90	144

5.2 Перечень лабораторных работ

Неделя семестра	Наименование лабораторной работы	Объем часов	Виды контроля
1-3	Работа в интегрированных средах разработки: -настройка профиля для последующей работы в IDE среде; -подготовка проектов с помощью готовых шаблонов	6	Отчет и защита
4-6	Базовые конструкции и ключевые слова языка программирования высокого уровня: -изучение стандартных команд и инструкций алгоритмического языка программирования; -использование средств ввода-вывода на экран ЭВМ.	6	Отчет и защита
7-9	Объектно-ориентированный подход к разработке программных средств: -создание расчетного приложения с описанием объектов в классе и инициализацией параметров; -создание проектов с динамически-изменяемыми структурами данных с освобождением памяти ЭВМ в процессе работы приложения; -использование статических переменных для обмена информацией между классами в разрабатываемом проекте.	6	Отчет и защита
10-13	Разработка приложений с графическими интерфейсными возможностями: -создание инструментальных панелей графического приложения; -использование ресурсов, работа с диалогами, организация всплывающих подсказок, интерактивных справочных данных и т.д.	6	Отчет и защита
14-15	Создание подсистем на основе подключаемых библиотек: -использование динамически-подключаемых библиотек; -использование статических ресурсов, а также иных средств, подготовленных сторонними разработчиками для заданной интегрированной среды разработки ПО.	6	Отчет и защита
16-18	Разработка кроссплатформенных приложений: -разработка приложений, работающих в операционных системах под мобильные платформы; -работа с директивами предпроцессора, конфигурирование проекта под запуск приложения в заданной ОС.	6	Отчет и защита

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 2 семестре для очной формы обучения.

Примерная тематика курсового проекта: «Расчетные задачи с применением графического интерфейса»

Задачи, решаемые при выполнении курсового проекта:

- расчетные задачи;
- простейшие геометрические преобразования;
- построение деревьев, введение в графы;
- сортировка данных;
- программирование математических выражений и структур;
- численные методы;
- вывод графических объектов на экран (графики функций, простейшие геометрические объекты);
- файловые операции;
- работа со строковыми и табличными данными;
- создание многооконных кроссплатформенных приложений;
- интеграция со сторонними приложениями.

Курсовой проект включает в себя графическую часть и расчетно-пояснительную записку.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ОПК-1	знать алгоритмы, методики подготовки к работе информационных систем, их типы.	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь поддерживать работоспособность информационных систем	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть приемами тестирования технических средств	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-6	знать основные характеристики аппаратных средств ЭВМ информационных	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

	систем			
	уметь осуществлять их настройку и выполнять конфигурирование указанных систем	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть приемами тестирования технических средств	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 2 семестре для очной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ОПК-1	знать алгоритмы, методики подготовки к работе информационных систем, их типы.	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь поддерживать работоспособность информационных систем	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть приемами тестирования технических средств	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ОПК-6	знать основные характеристики аппаратных средств ЭВМ информационных систем	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь осуществлять их настройку и выполнять конфигурирование указанных систем	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть приемами тестирования	Решение прикладных	Задачи решены в	Продемонстрирован	Продемонстрирован верный	Задачи не решены

	технических средств	задач в конкретной предметной области	полном объеме и получены верные ответы	верный ход решения всех, но не получен верный ответ во всех задачах	ход решения в большинстве задач	
--	---------------------	---------------------------------------	--	---	---------------------------------	--

7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Тип данных `int` представляет собой:

- а) целочисленный тип
- б) строковый тип
- в) вещественный тип
- г) логический тип

2. Ключевое слово `for`:

- а) означает использование в программе конструкции повторения
- б) означает использование в программе конструкции с условием
- в) слово не является ключевым

3. `bool flag=false`:

- а) объявлена и задана строка
- б) объявлена и задана логическая переменная
- в) объявлена и задана логическая константа

4. `main()`:

- а) точка входа в приложение
- б) заголовок функции
- в) утверждения в предыдущих ответах верны

5. Применение условий в составе программы требуется:

- а) когда необходимо сравнить несколько значений
- б) когда надо присвоить результат переменной
- в) когда программа запускается

6. Указатель - это:

- а) адрес в памяти
- б) ключевое слово
- в) переменная, заданная именем

7. Директивы предпроцессора:

- а) расширяют возможности приложения
- б) обеспечивают особые варианты сборки приложения
- в) все ответы правильные

8. Интегрированная среда разработки содержит:

- а) средства отладки
- б) средства авто дополнения кода
- в) средства анализа памяти
- г) все ответы правильные

9. Структуры позволяют:

- а) группировать данные по заданному признаку
- б) используются для хранения разнотипных данных
- в) обеспечивать взаимосвязь между записями
- г) все вышеперечисленное

10. Вещественные типы данных *float* и *double* отличаются:

- а) размером хранения данных
- б) это одно и то же
- в) не используются при создании программ

11) Класс -это:

- а) абстракция, описывающая методы, свойства, ещё не существующих объектов
- б) способ задать значение переменной по-другому
- в) объект структуры

12) Наследование в классах позволяет выполнить реализацию задачи:

- а) с использованием функций-шаблонов
- б) с использованием уже имеющихся классов
- в) с использованием переменных

7.2.2 Примерный перечень заданий для решения стандартных задач

№	Задача
1	Дана непустая последовательность неотрицательных целых чисел, оканчивающаяся отрицательным числом. Найти среднее арифметическое всех чисел последовательности (без учета отрицательного числа).
2	Среди чисел 1, 4, 9, 16, 25, ... найти первое число, большее n .
3	Дана непустая последовательность целых чисел, оканчивающаяся нулем. Найти: а) сумму всех чисел последовательности, больших числа x ; б) количество всех четных чисел последовательности
4	Дана последовательность ненулевых целых чисел. Определить, сколько раз в этой последовательности меняется знак. Например, в последовательности 10, -4, 12, 56, -4 знак меняется 3 раза.
5	Найти количество делителей каждого из целых чисел от 120 до

	140.
6	Дано слово. Получить и вывести на экран буквосочетание, состоящее из его третьего и последнего символа.
7	Дано слово. Получить его часть, образованную второй, третьей и четвертой буквами
8	Найти периметр треугольника, заданного координатами своих вершин. (Определить функцию для расчета длины отрезка по координатам его вершин.)
9	Дан двумерный массив. Найти: а) сумму элементов, расположенных в левом верхнем и правом нижнем углах; б) среднее арифметическое элементов, расположенных в четырех углах
10	Имеется файл, в котором записаны 12 вещественных чисел. Переписать все числа файла в массив в том же порядке.

7.2.3 Примерный перечень заданий для решения прикладных задач

№	Задача
1	Разработка текстового редактора
2	Разработка табличного редактора
3	Разработка архиватора
4	Разработка файлового менеджера
5	Разработка векторного редактора
6	Разработка графического редактора
7	Разработка приложения по просмотру графических форматов
8	Разработка приложения по просмотру 3D моделей
9	Разработка приложения по анализу аппаратной части
10	Разработка приложения по тестированию оборудования ЭВМ

7.2.4 Примерный перечень вопросов для подготовки к зачету

Не предусмотрено учебным планом

7.2.5 Примерный перечень вопросов для подготовки к экзамену

1. Структура и синтаксис простейшего консольного приложения.
2. Консольный ввод-вывод, использование функций из математической библиотеки.
3. Типы данных. Пример консольного приложения
4. Беззнаковые целочисленные типы данных, представление в памяти.
5. Вещественные типы данных, их представление в памяти.
6. Символьный тип данных, представление символов в памяти.
7. Преобразование типов данных.
8. Арифметические и логические операции. Приоритеты операций.
9. Битовые операции.
10. Переменные-константы.
11. Условный оператор if ... else. Примеры
12. Оператор выбора switch.

13. Оператор цикла *for*.
14. Оператор цикла *do-while*.
15. Оператор цикла *while*.
16. Вспомогательные операторы: *break, continue*.
17. Строковые переменные.
18. Стандартные функции для работы со строками.
19. Операция присваивания и ее разновидности.
20. Префиксное и постфиксное использование операций.
21. Понятие массивов. Одномерные массивы.
22. Понятие массивов. Многомерные массивы.
23. Указатели. Описание указателей.
24. Арифметические операции с указателями.
25. Ссылки.
26. Операторы выделения и освобождения памяти для переменной и массива.
27. Функции. Область видимости в функциях.
28. Передача переменных и массивов в функцию и возврат значений.
29. Использование функций с параметрами по умолчанию.
30. Перегруженные функции и функции-шаблоны.
31. Рекурсивные функции.
32. Понятие структур. Описание структур.
33. Перечисления. Примеры использования.
34. Последовательные контейнеры: стеки и очереди
35. Последовательные контейнеры: списки
36. Последовательные контейнеры: вектора
37. Контейнеры типа множества (*set*) и ассоциативные списки (*map*)
38. Применение исключений в разработке программ
39. Заголовочные файлы.
40. Понятие препроцессора. Назначение директив препроцессоров.
41. Введение в классы и объекты.
42. Конструкторы и деструкторы.
43. Спецификаторы доступа. Методы *set* и *get* для доступа к закрытым элементам класса.
44. Неявный указатель класса *this*. Примеры использования.
45. Введение в наследование классов.

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Экзамен проводится по билетам, каждый из которых содержит 2 вопроса и 2 задачи. Каждый правильный ответ на вопрос оценивается 5 баллами, задача оценивается в 5 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20. Для получения зачета по дисциплине необходимо набрать минимум 10 баллов.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.
2. Оценка «Удовлетворительно» ставится в случае, если студент

набрал от 6 до 10 баллов

3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов.)

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Интегрированные среды и средства разработки на языке программирования высокого уровня	ОПК-1, ОПК-6	Тест, защита лабораторных работ, требования к курсовому проекту, экзамен
2	Базовые конструкции и ключевые слова языка программирования высокого уровня	ОПК-1, ОПК-6	Тест, защита лабораторных работ, требования к курсовому проекту, экзамен
3	Объектно-ориентированный подход к разработке программных средств	ОПК-1, ОПК-6	Тест, защита лабораторных работ, требования к курсовому проекту, экзамен
4	Разработка приложений с графическими интерфейсными возможностями	ОПК-1, ОПК-6	Тест, защита лабораторных работ, требования к курсовому проекту, экзамен
5	Создание подсистем на основе подключаемых библиотек	ОПК-1, ОПК-6	Тест, защита лабораторных работ, требования к курсовому проекту, экзамен
6	Построение кроссплатформенных приложений	ОПК-1, ОПК-6	Тест, защита лабораторных работ, требования к курсовому проекту, экзамен

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и аттестация выставляется, если количество правильных ответов более 65%.

Решение стандартных задач осуществляется на компьютере с предустановленной средой программирования. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется

оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, осуществляется на компьютере с предустановленной средой программирования. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Защита курсовой работы, курсового проекта или отчета по всем видам практик осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

1. Павловская Т.А.С/С++. Процедурное и объектно-ориентированное программирование [Текст]: учебник : допущено Министерством образования и науки Российской Федерации. - Москва; Санкт-Петербург; Нижний Новгород [и др.] : Питер, 2017 (СПб. : Первая Академ. тип."Наука", 2014). - 495 с.

2. Сундукова Т.О. Структуры и алгоритмы компьютерной обработки данных [Электронный ресурс]/ Сундукова Т.О., Ваныкина Г.В. –Электрон.текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2011. – 475 с. – Режим доступа: <http://www.iprbookshop.ru/16736>.

3. Орлов С.А. Теория и практика языков программирования [Текст] : учебник. - Москва ; Санкт-Петербург ; Нижний Новгород [и др.] : Питер, 2014 (Чехов : Первая Образцовая тип., фил. "Чеховский Печатный Двор", 2014). - 688 с.

4. Ахо А.В. Структура данных и алгоритмы [Текст] = Datastructuresandalgorithms: учебное пособие / пер. с англ. и ред. А. А. Минько. - М.; СПб.; Киев: Вильямс, 2008. - 382

5. Вирт Н. Алгоритмы и структуры данных. Новая версия для Оберона [Электронный ресурс]/ Никлаус Вирт— Электрон. текстовые данные.— М.: ДМК Пресс, 2010.— 272 с.— Режим доступа: <http://www.iprbookshop.ru/7965>.

6. Каширин И.Ю. От С к С++ [Электронный ресурс]: учебное пособие/ Каширин И.Ю., Новичков В.С.— Электрон.текстовые данные.— М.: Горячая линия - Телеком, 2012.— 334 с.— Режим доступа: <http://www.iprbookshop.ru/12022>.

7. Синюк В.Г. Алгоритмы и структуры данных [Электронный ресурс]: лабораторный практикум. Учебное пособие/ Синюк В.Г., Рязанов Ю.Д.— Электрон.текстовые данные.— Белгород: Белгородский государственный технологический университет им. В.Г. Шухова, ЭБС АСВ,

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

Программное обеспечение

Среда разработки MICROSOFT VISUAL STUDIO 2010 и выше.

Информационно-справочные системы

1. Сайт ixbt.com www.ixbt.com Полная оперативная и объективная информация о персональных компьютерах, их компонентах и периферийных устройствах

2. Сайт CITForum www.citforum.ru Библиотека технических материалов по информационным технологиям

3. Сайты поддержки разработчиков ПО msdn.microsoft.com. Справочная техническая документация Microsoft по среде разработки Visual Studio и поддержки языков программирования

4. Комитет по стандартизации в области радиоэлектроники и вычислительной техники www.ieee.org. Нормативно-справочная документация по вычислительной технике

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Компьютерный класс с предустановленной средой разработки приложений MICROSOFT VISUAL STUDIO 2010 и выше.

Мультимедийные средства: наборы файлов презентаций по темам лекционных занятий, комплект видеороликов по установке, настройке и примерам использования инструментальных средств технологии программирования.

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Среды объектно-ориентированного программирования» читаются лекции, проводятся лабораторные работы, выполняется курсовой проект.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на лабораторном оборудовании в

соответствии с методиками, приведенными в указаниях к выполнению работ.

Методика выполнения курсового проекта изложена в учебно-методическом пособии. Выполнять этапы курсового проекта должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины производится проверкой курсового проекта, защитой курсового проекта.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом, экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.