

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Филиал федерального государственного бюджетного образовательного
учреждения высшего образования
«Воронежский государственный технический университет»
в городе Борисоглебске



УТВЕРЖДАЮ

Директор филиала

/В.В. Григораш/

34 августа 2021 г.

**РАБОЧАЯ ПРОГРАММА
дисциплины**

«Объектно-ориентированное программирование»

Направление подготовки 09.03.02 Информационные системы и технологии

Профиль Информационные системы и технологии цифровизации

Квалификация выпускника бакалавр

Нормативный период обучения 4 г

Форма обучения очная

Год начала подготовки 2021

Автор программы

В.В. Волков

Заведующий кафедрой
естественнонаучных дисциплин

Л.И. Матвеева

Руководитель ОПОП

Е.А. Позднова

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины

Изучение объектно-ориентированной методологии программирования, изучение основных понятий объектно-ориентированного программирования.

1.2. Задачи освоения дисциплины

- к теоретическим задачам относятся: обучение разработке программного обеспечения в объектно-ориентированной парадигме; изучение основ программирования на языке C++;

- прикладные задачи состоят в приобретении навыков проектирования и разработки программного обеспечения с применением объектно-ориентированного подхода к программированию.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Объектно-ориентированное программирование» относится к дисциплинам обязательной части блока Б.1 учебного плана.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Объектно-ориентированное программирование» направлен на формирование следующих компетенций:

ОПК-2 - Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;

ОПК-4 - Способен участвовать в разработке технической документации, связанной с профессиональной деятельностью с использованием стандартов, норм и правил;

ОПК-6 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий;

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-2	Знать методы и технологии программирования, принципы и определения объектно-ориентированного подхода
	Уметь описывать основные структуры данных на основе объектно-ориентированного подхода
	Владеть навыками работы в различных средах программирования
ОПК-4	Знать виды технической документации поддержки программного кода
	Уметь описывать техническую документацию

	Владеть навыками оформления программного кода
ОПК-6	Знать основные принципы конструирования алгоритмов
	Уметь разрабатывать и реализовывать алгоритмы на языке высокого уровня
	Владеть методами описания основных классов и алгоритмов на их основе.

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Объектно-ориентированное программирование» составляет 6 зачетных единиц.

Распределение трудоемкости дисциплины по видам занятий

Очная форма обучения

Вид учебной работы	Всего часов	Семестры
		2
Аудиторные занятия (всего)	90	90
В том числе:		
Лекции	36	36
Практические занятия (ПЗ)	-	-
Лабораторные работы (ЛР)	54	54
Самостоятельная работа	90	90
Курсовой проект (работа)	+	+
Контрольная работа	-	-
Вид промежуточной аттестации (Экзамен)	36	36
Общая трудоемкость час	216	216
зач. ед.	6	6

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

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	Общие сведения, Основы программирования на С++	Введение. Необходимость объектно-ориентированное программирование, характеристики объектно-ориентированных языков Основы программирования на С++. История С++, сравнение с языком С. Структура программы на языке С++. Директивы, комментарии. Переменные. Преобразование типов. Арифметические операции. Обзор интегрированных сред разработки (IDE).	8	12	18	38
2	Циклы и ветвления, структуры и функции	Циклы и ветвления. Операции отношения. Циклы и ветвления. Логические операции. Приоритеты операций. Структуры. Использование структур, перечислений. Функции в С++	8	12	18	38

3	Объектно-ориентированный подход к разработке программных средств	Введение в классы и объекты. Описание и свойства объектов. Конструкторы, деструкторы классов, объявления статических объектов, переменных и методов в классе. Наследование в ООП. Наследование, виды наследования. Абстрактные классы. Множественное наследование.	8	10	18	36
4	Потоки и файлы. Многофайловые программы.	Потоки и файлы. Потокосовые классы. Потокосовый ввод/вывод дисковых файлов. Указатели файлов. Файловый ввод/вывод с помощью методов. Многофайловые программы. Создание многофайловой программы. Межфайловое взаимодействие.	6	10	18	34
5	Шаблоны и исключения. Стандартная библиотека шаблонов.	Шаблоны и исключения. Шаблоны функций и классов. Исключения. Стандартная библиотека шаблонов. Последовательные контейнеры. Итераторы. Специальные итераторы. Ассоциативные контейнеры. Хранение пользовательских объектов. Функциональные объекты	6	10	18	34
Итого			36	54	90	180

5.2 Перечень лабораторных работ

Лабораторная работа № 1. Работа с компилятором g++ и IDE Anjuta в ОС Linux.

Лабораторная работа № 2. Изучение базовых инструкций языка C++. Особенности работы с массивами в языке C++.

Лабораторная работа № 3. Использование встроенных типов данных языка C++. Манипулирование основными системами счисления.

Лабораторная работа № 4. Символьные и строковые типы данных в языке C++. Основные функции работы со строками.

Лабораторная работа № 5. Работа с файлами в языке C++. Запись и чтение информации из файла.

Лабораторная работа № 6. Введение в понятие класса в языке C++.

Лабораторная работа № 7. Динамические структуры данных. Организация списка с помощью ООП.

Лабораторная работа № 8. Изучение концепций инкапсуляции, наследования и полиморфизма.

Лабораторная работа № 9. Изучение основных контейнеров и алгоритмов стандартной библиотеки шаблонов (STL).

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

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта во 2 семестре.

Примерная тематика курсового проекта: «Разработка программного обеспечения для приемной комиссии ВУЗа с использованием объектно-ориентированного подхода».

Задачи, решаемые при выполнении курсового проекта. Требуется разработать программную систему, предназначенную для работников приемной комиссии высшего учебного заведения. Она должна обеспечивать хранение, просмотр и изменение сведений об абитуриентах, а также о расписании экзаменов и их результате. Результатом работы приемной комиссии должен быть список абитуриентов, зачисленных в институт.

Курсовой проект включает в себя графическую часть и расчетно-

пояснительную записку.

Учебным планом по дисциплине «Объектно-ориентированное программирование» предусмотрено выполнение контрольной работы (контрольных работ) во 2 семестре.

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-2	Знать методы и технологии программирования, принципы и определения объектно-ориентированного подхода	Тестирование Проверка этапов выполнения курсового проектами Результаты проведения коллоквиума Ответы на теоретические вопросы при защите лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Уметь описывать основные структуры данных на основе объектно-ориентированного подхода	Эффективность использования изученного теоретического материала при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Владеть навыками работы в различных средах программирования	Владение языком программирования C++ при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-4	Знать виды технической документации поддержки программного кода	Тестирование Проверка этапов выполнения курсового проектами Результаты проведения коллоквиума Ответы на теоретические вопросы при защите лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Уметь описывать техническую документацию	Эффективность использования изученного теоретического материала при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Владеть навыками оформления программного кода	Владение языком программирования C++ при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

ОПК-6	Знать основные принципы конструирования алгоритмов	Тестирование Проверка этапов выполнения курсового проекта Результаты проведения коллоквиума Ответы на теоретические вопросы при защите лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Уметь разрабатывать и реализовывать алгоритмы на языке высокого уровня	Эффективность использования изученного теоретического материала при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Владеть методами описания основных классов и алгоритмов на их основе.	Владение языком программирования C++ при выполнении лабораторных работ	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 во 2 семестре для очной формы обучения по системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ОПК-2	Знать методы и технологии программирования, принципы и определения объектно-ориентированного подхода	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	Уметь описывать основные структуры данных на основе объектно-ориентированного подхода	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 навыками работы в различных средах программирования	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-4	Знать виды технической документации поддержки программного кода	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов

	Уметь описывать техническую документацию	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 навыками оформления программного кода	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 принципы конструирования алгоритмов	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	Уметь разрабатывать и реализовывать алгоритмы на языке высокого уровня	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 методами описания основных классов и алгоритмов на их основе.	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

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. Дополнительная информация, передающаяся при генерации исключения, может быть помещена в:

- A. ключевое слово `throw`;
- B. функцию, вызвавшую ошибку;
- C. блок-ловушку;
- D. объект класса исключений.

2. Чтобы определять объекты класса в разных файлах, в каждом из них необходимо:

- A. объявлять класс;
- B. определять класс;
- C. объявлять класс с использованием `extern`;
- D. определять класс с использованием `extern`.

3. Оператор присваивания может быть перегружен с целью:

- A. хранения информации о количестве одинаковых объектов;
- B. присваивания идентификационного номера каждому объекту;

C. проверки того, что все компонентные данные скопировались без ошибок;

D. уведомления о том, что имело место присваивание.

4. Истинно ли утверждение о том, что указатель на базовый класс может ссылаться на объекты порожденного класса? А. да; В. нет. 5. Оператор разрешения обычно:

A. ограничивает видимость переменных для определенных методов;

B. обозначает, от какого базового класса создан производный;

C. определяет отдельный класс;

D. разрешает неопределенности.

7.2.2 Примерный перечень заданий для решения стандартных задач

1. Предположим, что класс X не использует перегруженные операции. Напишите выражение, в котором вычитается объект x1 класса X из другого объекта x2 этого же класса. Результат помещается в x3.

2. Истинно ли следующее утверждение: переменная типа char может хранить значение 301?

3. Опишите структуру, содержащую три переменные типа int с названиями hrs, mins и secs. Назовите структуру именем time.

4. Напишите функцию foo(), выводящую на экран слово foo.

5. Дана следующая функция: `int times2(int a) { return (a * 2); }` Напишите функцию main(), которая будет содержать все необходимое для вызова данной функции.

7.2.3 Примерный перечень заданий для решения прикладных задач

1. Истинно ли следующее утверждение: если вы определили конструктор, содержащий определение типа `aclass obj = intvar;`, вы также можете записать выражение типа `obj = intvar;`?

2. Пусть указатель p ссылается на объекты базового класса и содержит адрес объекта порожденного класса. Пусть в обоих этих классах имеется не виртуальный метод ding(). Тогда выражение `p->ding();` поставит на выполнение версию функции ding() из _____ класса.

3. Объясните разницу в выполнении следующих двух выражений: `person p1(p0);` `person p1 = p0;`

4. Напишите if, определяющий, достиг объект ifstream под названием foobar конца файла или же возникла ошибка.

5. Напишите выражение, записывающее единичный символ в объект fileOut класса ofstream.

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. Виды переменных в C++

2. Понятие класса в C++. Применение конструкторов и деструкторов

класса.

3. Работа с массивами в C++.
4. Области видимости переменных и методов в классе.
5. Наследование в C++. Примеры простого и множественного наследования.
6. Указатели на объекты.
7. Работа с потоковыми классами в C++.
8. Поточковый ввод/вывод файлов.
9. Компиляция программы на C++ с использованием консоли и компилятора g++.
10. Стандартная библиотека шаблонов (STL)

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Например: Экзамен проводится по тест-билетам, каждый из которых содержит 10 вопросов и задачу. Каждый правильный ответ на вопрос в тесте оценивается 1 баллом, задача оценивается в 10 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20.

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	Общие сведения, Основы программирования на C++	ОПК-2, ОПК-4, ОПК-6	Тест, экзамен, защита лабораторных работ, защита курсового проекта
2	Циклы и ветвления, структуры и функции	ОПК-2, ОПК-4, ОПК-6	Тест, экзамен, защита лабораторных работ, защита курсового проекта
3	Объектно-ориентированный подход к разработке программных средств	ОПК-2, ОПК-4, ОПК-6	Тест, экзамен, защита лабораторных работ, защита курсового проекта
4	Потоки и файлы. Многофайловые программы	ОПК-2, ОПК-4, ОПК-6	Тест, экзамен, защита лабораторных работ, защита курсового проекта
5	Шаблоны и исключения. Стандартная библиотека шаблонов.	ОПК-2, ОПК-4, ОПК-6	Тест, экзамен, защита лабораторных работ, защита курсового проекта

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 выставляется оценка согласно методике выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике 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. Язык программирования С++ для профессионалов [Электронный ресурс] / Б. Страуструп. - Язык программирования С++ для профессионалов; 2021-01-23. - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 670 с. - Лицензия до 23.01.2021. - ISBN 2227-8397.

URL: <http://www.iprbookshop.ru/73737.html>

2. Холопкина, Л.В. Программирование на языке С++: Учеб. пособие. - Воронеж: ГОУВПО "Воронежский государственный технический университет", 2011. - 184 с. - 250 экз.; 171-97.

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:

Перечень ПО, включая перечень лицензионного программного обеспечения:

Microsoft Office 64-bit;

ОС Windows 7 Pro;

Mozilla Firefox 81.0 (x64 ru);

Google Chrome;
PDFCreator;
Microsoft .NET Framework;
PascalABC.NET;
Code: Blocks;
Python;
Microsoft Visual Studio Code
Ресурсы информационно-телекоммуникационной сети «Интернет»:
<http://window.edu.ru>
<http://www.edu.ru/>
Образовательный портал ВГТУ

Современные профессиональные базы данных и информационно-справочные системы:

<https://proglib.io/>
<https://habr.com/ru/>
www.consultant.ru

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Для проведения обучения по дисциплине используется компьютерный класс.

Аудитория 6

Компьютерный класс

Учебная аудитория для проведения учебных занятий

Комплект учебной мебели:

- рабочее место преподавателя (стол, стул);
- рабочие места обучающихся (столы, стулья)

- персональные компьютеры с установленным ПО, подключенные к сети Интернет (12 шт.);
- принтер;
- доска магнитно-маркерная поворотная

Аудитория 7

Компьютерный класс

Учебная аудитория для проведения учебных занятий

Комплект учебной мебели:

- рабочее место преподавателя (стол, стул);
- рабочие места обучающихся (столы, стулья)

- персональные компьютеры с установленным ПО, подключенные к сети Интернет (11 шт.);
- принтер цветной лазерный;
- доска магнитно-маркерная поворотная

Помещение для самостоятельной работы. Читальный зал с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

По дисциплине «Объектно-ориентированное программирование» 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.
Практическое занятие	Конспектирование рекомендуемых источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы. Прослушивание аудио- и видеозаписей по заданной теме, выполнение расчетно-графических заданий, решение задач по алгоритму.

<p>Самостоятельная работа</p>	<p>Самостоятельная работа студентов способствует глубокому усвоения учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие:</p> <ul style="list-style-type: none"> - 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.
<p>Подготовка к промежуточной аттестации</p>	<p>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 зачетом, зачетом с оценкой три дня эффективнее всего использовать для повторения и систематизации материала.</p>

