

Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Курганский государственный университет»
(КГУ)

Кафедра «Механика машин и основы конструирования»



Утверждаю
Ректор
/Н.В. Дубив/
2020 г.

Рабочая программа учебной дисциплины

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

образовательной программы высшего образования –
программы специалитета
01.05.01 – Фундаментальные математика и механика

Направленность:
Математическое и компьютерное моделирование механических систем

Формы обучения: очная

Курган 2020

Рабочая программа дисциплины «Объектно-ориентированное программирование» составлена в соответствии с учебным планом по программе специалитета Фундаментальные математика и механика (Математическое и компьютерное моделирование механических систем), утвержденным для очной формы обучения 28.08.2020 г.

Рабочая программа дисциплины одобрена на заседании кафедры «Механика машин и основы конструирования» 11.09.2020 г., протокол № 1.

Рабочую программу составил

к.т.н., доцент



Д.А. Курасов

Согласовано:

Заведующий кафедрой

«Механика машин и основы конструирования»



Д.А. Курасов

Специалист по учебно-методической работе
Учебно-методического отдела



Г.В. Казанкова

1. ОБЪЕМ ДИСЦИПЛИНЫ

Всего: 3 зачетных единицы трудоемкости (108 академических часов)

Очная форма обучения

Вид учебной работы	На всю дисциплину	Семестр
		7
Аудиторные занятия (контактная работа с преподавателем), всего часов	48	48
в том числе:		
Лекции	16	16
Лабораторные работы	32	32
Самостоятельная работа, всего часов	60	60
в том числе:		
Подготовка к зачету	18	18
Другие виды самостоятельной работы (самостоятельное изучение тем (разделов) дисциплины)	42	42
Вид промежуточной аттестации	Зачет	Зачет
Общая трудоемкость дисциплины и трудоемкость по семестрам, часов	108	108

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина «Объектно-ориентированное программирование» относится к части, формируемой участниками образовательных отношений, блока 1.

Изучение дисциплины «Объектно-ориентированное программирование» играет важную роль в подготовке специалиста.

Освоение обучающимися дисциплины «Объектно-ориентированное программирование» опирается на знания, умения, навыки и компетенции, приобретенные в результате освоения предшествующих дисциплин:

- Программирование и работа на ЭВМ;
- Основы проектирования и компьютерные технологии.

Результаты обучения по дисциплине необходимы для освоения последующих дисциплин:

- Компьютерное моделирование механических систем;
- Теория управления и оптимизации;
- Научно-исследовательская работа.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Целью освоения дисциплины «Объектно-ориентированное программирование» является: ознакомление студентов с основными возможностями и особенностями применения языка программирования C++, а также получение практических навыков работы с этим языком.

Задачами освоения дисциплины «Объектно-ориентированное программирование» является изучение основных понятий объектно-ориентированного программирования, изучение основ языка программирования C++, использование этого языка для решения практических задач.

Компетенции, формируемые в результате освоения дисциплины:

- способность использовать методы компьютерного моделирования при решении прикладных задач исследования сложных физических процессов в твердых телах, жидкостях и газах (ПК-3);
- способность разрабатывать физические и компьютерные модели производственно-технологических объектов, сред и конструкций, а также использовать современное экспериментальное оборудование (ПК-6).

В результате изучения дисциплины обучающийся должен знать:

- методы математического и алгоритмического моделирования при решении прикладных задач (ПК-3);
- основные этапы решения задач с использованием современных программных средств и технологий (ПК-6).

Обучающийся должен уметь:

- использовать изученные инструментальные средства для решения практических задач профессиональной деятельности (для ПК-3);
- создавать простейшие приложения, используя язык программирования C++ (для ПК-3, ПК-6);
- анализировать поставленную задачу и проводить ее декомпозицию (для ПК-6).

Обучающийся должен владеть:

- технологией решения типовых информационных и вычислительных задач с помощью прикладных программ (для ПК-6);
- навыками поэтапной разработки программных приложений (для ПК-3, ПК-6).

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Учебно-тематический план

Рубеж	Номер раздела, темы	Наименование раздела, темы	Количество часов контактной работы с преподавателем	
			Лекции	Лабораторные работы
7 семестр				
Рубеж 1	1	Общие сведения о программах, лексемах и алфавите	2	–
	2	Основные управляющие конструкции C++	2	6
	3	Массивы в C++	2	4
	4	Функции в C++	2	–
	5	Строки в C++	2	4
			Рубежный контроль №1	
Рубеж 2	6	Работа с файлами в C++	2	6
	7	Структуры в C++	2	4
	8	Объединения в C++	2	4
			Рубежный контроль №2	
Всего:			16	32

4.2. Содержание лекционных занятий

Тема 1. Общие сведения о программах, лексемах и алфавите

Лексические основы языка C++. Алфавит. Лексемы. Скалярные типы и выражения. Использование простейших типов. Функции ввода и вывода. Операции. Знаки операций. Арифметические операции. Инкремент и декремент. Операции присваивания и отношения. Логические операции. Условная операция. Операции преобразования типов.

Тема 2. Основные управляющие конструкции C++

Условные конструкции. Оператор if...else. Оператор switch. Циклические конструкции. Цикл while. Цикл do...while. Цикл for. Оператор безусловного перехода, оператор принудительного выхода из цикла или переключателя. Оператор завершения выполнения текущего шага тела цикла. Оператор возвращения значения из функции.

Тема 3. Массивы в C++

Массивы. Общие сведения о массивах. Одномерные массивы. Имя массива. Двумерные массивы. Многомерные массивы, работа с массивами.

Тема 4. Функции в C++

Функции. Общие сведения о функциях. Определение функции. Аргументы функций. Передача нескольких значений в функцию. Возврат нескольких значений из функции. Функции с переменным числом параметров. Рекурсия.

Тема 5. Строки в C++

Строки. Строковая константа. Инициализация строк. Функции для работы со строками.

Тема 6. Работа с файлами в C++

Работа с файлами. Функции высокоуровневого ввода/вывода. Функции низкоуровневого ввода/вывода.

Тема 7. Структуры в C++

Структуры. Описание структур. Операции над структурами. Использование полей битов в качестве полей структур.

Тема 8. Объединения в C++

Объединения. Понятие объединения. Совместное использование объединений и структур. Переменные структур.

4.3. Лабораторные работы

Номер раздела, темы	Наименование раздела, темы	Наименование лабораторной работы	Норматив времени, час
2	Основные управляющие конструкции C++	Составление линейных программ на языке C++	2
		Составление разветвляющихся программ на языке C++	2
		Составление циклических программ на языке C++	2
3	Массивы в C++	Одномерные массивы в языке C++	2
		Многомерные массивы в языке C++	2
5	Строки в C++	Обработка строковой информации	4
	Рубежный контроль №1		2
6	Работа с файлами в C++	Решение задач с использованием функций высокоуровневого ввода/вывода	4
		Решение задач с использованием функции низкоуровневого ввода/вывода	2
7	Структуры в C++	Использование структур при решении задач	4
8	Объединения в C++	Использование объединений при решении задач	4
	Рубежный контроль №2		2
Всего:			32

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

При прослушивании лекций рекомендуется отмечать в конспектах все важные моменты, касающиеся используемых при написании программы команд и функций, в частности тех, которые направлены на качественное выполнение лабораторных работ.

Залогом качественного выполнения лабораторных работ является самостоятельная подготовка к ним накануне путем повторения материалов предыдущих лекционных и лабораторных занятий. Рекомендуется подготовить вопросы по неясным моментам и обсудить их с преподавателем в начале лабораторной работы.

В преподавании дисциплины применяются образовательные технологии: метод проблемного изложения материала; самостоятельное ознакомление студентов с источниками информации, использование иллюстративных материалов (компьютерные презентации), демонстрируемых на современном оборудовании, общение в интерактивном режиме.

Самостоятельная работа студента, наряду с лабораторными аудиторными занятиями в группе выполняется (при непосредственном или опосредованном контроле преподавателя) по учебникам и учебным пособиям, оригинальной современной литературе по профилю.

Для текущего контроля успеваемости по очной форме обучения преподавателем используется балльно-рейтинговая система контроля и оценки академической активности. Поэтому настоятельно рекомендуется тщательно прорабатывать материал дисциплины при самостоятельной работе, участвовать во всех формах обсуждения и взаимодействия как на лекциях, так и на лабораторных занятиях в целях лучшего освоения материала и получения высокой оценки по результатам освоения дисциплины.

Выполнение самостоятельной работы подразумевает самостоятельное изучение разделов дисциплины, подготовку к лабораторным работам, рубежным контролям, подготовку к зачету.

Рекомендуемая трудоемкость самостоятельной работы представлена в таблице:

Рекомендуемый режим самостоятельной работы

Наименование вида самостоятельной работы	Рекомендуемая трудо- емкость, акад. час.
	7 семестр
Самостоятельное изучение тем дисциплины:	22
Ортогональные списки. Операции над ортогональными списками.	4
Списки магазинного типа. Очереди. Формирование очереди. Добавление звена к очереди. Удаление звена из очереди.	4
Списки магазинного типа. Стеки. Формирование стека. Добавление звена в стек. Удаление звена из стека.	4
Списки магазинного типа. Деки. Формирование дека. Добавление элемента в дек. Удаление элемента из дека	4
Линейные двунаправленные списки. Двунаправленные кольцевые списки. Деки на основе двунаправленных списков.	6
Подготовка к лабораторным занятиям (по 1 часу на каждое занятие)	16
Подготовка к рубежным контролям (по 2 часа на каждый рубеж)	4
Подготовка к зачету	18
Всего:	60

Приветствуется выполнение разделов самостоятельной работы в компьютерном классе кафедры «Гусеничные машины и прикладная механика».

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

6.1. Перечень оценочных средств

1. Балльно-рейтинговая система контроля и оценки академической активности студентов в КГУ.
2. Перечень вопросов к рубежным контролям №1, №2.
3. Перечень заданий к зачету.

6.2. Система балльно-рейтинговой оценки работы студентов по дисциплине

№	Наименование	Содержание						
		Распределение баллов за 7 семестр						
1	Распределение баллов за семестры по видам учебной работы, сроки сдачи учебной работы (доводятся до сведения студентов на первом учебном занятии)	Вид учебной работы:	Посещение лекций	Выполнение и защита отчетов по лабораторным работам	Рубежный контроль 1	Рубежный контроль 2	Зачет	
		Балльная оценка:	до 8	до 52	до 5	до 5		до 30
		Примечания	8 лекций по 1 баллу	4 балла за 2-х часовую работу – 24 балла; 7 баллов за 4-х часовую работу – 28 баллов	на 8-м лабораторном занятии	на 16-м лабораторном занятии		
2	Критерий пересчета баллов в традиционную оценку по итогам работы в семестре и зачета	60 и менее баллов – неудовлетворительно; незачтено 61...73 – удовлетворительно; зачтено 74... 90 – хорошо; 91...100 – отлично						
3	Критерий допуска к промежуточной аттестации, возможности получения автоматического зачета по дисциплине, возможность получения бонусных баллов	<p>Для допуска к промежуточной аттестации (зачету) студент должен выполнить и защитить отчеты по всем лабораторным работам, набрать по итогам текущего и рубежного контроля не менее 50 баллов.</p> <p>Для получения зачета «автоматически» студенту необходимо набрать за семестр минимальное количество баллов - 61.</p> <p>По согласованию с преподавателем студенту могут быть добавлены дополнительные (бонусные) баллы за активное участие в научной и методической работе, оригинальность принятых решений в ходе выполнения лабораторных работ, за участие в значимых учебных и внеучебных мероприятиях кафедры и выставлен зачет «автоматически».</p>						
4	Формы и виды учебной работы для неуспевающих (восстановившихся на курсе обучения) студентов для получения недостающих баллов в конце семестра	<p>В случае если к промежуточной аттестации (зачету) набрана сумма менее 50 баллов, студенту необходимо набрать недостающее количество баллов за счет выполнения дополнительных заданий, до конца последней (зачетной) недели семестра. При этом необходимо проработать материал всех пропущенных лабораторных работ.</p> <p>Формы дополнительных заданий (назначаются преподавателем):</p> <ul style="list-style-type: none"> - выполнение и защита отчетов по пропущенным лабораторным работам (до 4 баллов); - прохождение рубежного контроля (до 5 баллов). <p>Ликвидация академических задолженностей, возникших из-за разности в учебных планах при переводе или восстановлении, проводится путем выполнения дополнительных заданий, форма и объем которых определяется преподавателем.</p>						

6.3. Процедура оценивания результатов освоения дисциплины

Рубежный контроль 1, 2 проходит в форме собеседования и заключается в ответе на два вопроса, задаваемые преподавателем. На подготовку к ответу отводится не менее 40 минут.

Перед проведением каждого рубежного контроля преподаватель прорабатывает со студентами основной материал соответствующих разделов дисциплины в форме краткой лекции-дискуссии.

Преподаватель оценивает в баллах результаты рубежных контролей 1, 2 и заносит в ведомость учета текущей успеваемости.

Билеты к зачету состоят из двух вопросов и практического задания. Ответы на каждый вопрос оцениваются до 10 баллов, выполнение практического задания оценивается до 10 баллов. Время, отводимое студенту на решение задачи, составляет 1 астрономический час.

Результаты текущего контроля успеваемости и зачета заносятся преподавателем в зачетную ведомость, которая сдается в организационный отдел института в день сдачи зачета, а также выставляются в зачетную книжку студента.

6.4. Примеры оценочных средств для рубежных контролей и зачета

Примерные задания к рубежному контролю № 1

1. Что напечатает следующая программа?

```
#include<iostream.h>
main (){
int x;
x = -3+4*5-6; cout<<x<<" /* Операция 1 */
x = 3+4%5-6; cout<<x<<" /* Операция 2 */
x = -3*4%-6/5; cout<<x<<" /* Операция 3 */
x = (7+6)%5/2; cout<<x<<" /* Операция 4 */
```

2. Что напечатает следующая программа.

```
#include<iostream.h>
#include<stdio.h>
main() {
ini a.b.c.v.k;
con 1 «"Задайте целое число:
cin»v: k=v:
cout<<" До Во время После"«endl:
v=k:a='\:b=v++:c=v;printf("v++%8d%8d%8d\n",a,b,c);
v=k:a=v;b=v~:c=v;printf("v— %8d%8d%8d\n",a,b,c);
v=k:a=v;b=++v;c=v;prmtf("-H-v%8d%8d%8d\n",a,b,c):
v=k:a=v;b=— v;c=v;printf("— v%8d%8d%8d\n",a,b,c); }
3. Проиллюстрируйте в программе применение логических операций и операций увеличения.
```


4. Выдать на печать в обратном порядке цифры целого положительного числа N.

Примерные задания к рубежному контролю № 2

1. Напишите программу для замены в слове X всех букв "а" на сочетание "ку".
2. Напишите программу, удваивающую каждую букву слова X.
3. Вычеркните из слова X буквы, стоящие на четных местах.
4. Составить массив, каждый элемент которого равен максимальному из соответствующих значений двух других массивов.

Примерный перечень вопросов к зачету

1. Алфавит языка C++, Лексемы языка C++.
2. Знаки операций в языке C++.
3. Скалярные типы и выражения в языке C++. Примеры использования простейших типов в языке C++.
4. Функции форматного ввода и вывода. Примеры.
5. Арифметические операции в языке C++. Примеры.
6. Инкремент и декремент. Примеры.
7. Операции присваивания и отношения. Примеры.
8. Логические операции. Примеры.
9. Условная операция. Примеры.
10. Операция sizeof. Примеры.
11. Операции преобразования типов. Примеры.
12. Оператор if...else. Примеры.
13. Оператор switch. Примеры.
14. Цикл while. Примеры.
15. Цикл do... while. Примеры.
16. Цикл Gog. Примеры.
17. Оператор безусловного перехода. Примеры.
18. Оператор принудительного выхода из цикла или переключателя. Примеры.
19. Оператор завершения выполнения текущего шага тела цикла. Примеры.
20. Оператор, возвращающий значение из функции. Примеры.
21. Одномерные массивы. Примеры.
22. Двумерные массивы. Примеры.
23. Определение функции. Примеры.
24. Аргументы функции. Примеры.
25. Передача нескольких значений в функцию. Возврат нескольких значений из функции. Примеры.
26. Рекурсия. Примеры.
27. Функции с переменным числом параметров. Примеры.
28. Подставляемые функции. Примеры.
29. Способы передачи массивов в функции и их возврата. Примеры.

30. Автоматический класс памяти (auto). Примеры.
31. Внешний класс памяти (extern). Примеры.
32. Внешние глобальные объекты (extern). Примеры.
33. Внешние статические объекты (static). Примеры.
34. Регистровый класс памяти (register). Примеры.
35. Строковая константа. Инициализация строк. Строки и указатели. Примеры.
36. Функции работы со строками. Примеры.
37. Использование строк в командной строке. Примеры.
38. Функции высокоуровневого ввода, вывода. Примеры.
39. Форматный ввод/вывод. Примеры.
40. Функции низкоуровневого ввода/вывода. Примеры.
41. Описание структур. Примеры.
42. Операции над структурами. Примеры.
43. Использование полей битов в качестве полей структур. Примеры.

6.5. Фонд оценочных средств

Полный банк заданий для текущего, рубежных контролей и промежуточной аттестации по дисциплине, показатели, критерии, шкалы оценивания компетенций, методические материалы, определяющие процедуры оценивания образовательных результатов, приведены в учебно-методическом комплексе дисциплины.

7. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА

7.1. Основная учебная литература

1. Фридман А. Л. Язык программирования Си++: курс лекций : учебное пособие : для студентов вузов, обучающихся по специальности 351400 "Прикладная информатика" / А.Л. Фридман : Интернет-Университет информационных технологий. – Изд. 2-е. испр. – Москва : Интернет-Университет информационных технологий. 2004. – 257 с.
2. Зыков С.В. Введение в теорию программирования: курс лекций : учебное пособие : для студентов вузов, обучающихся по специальности 351400 «Прикладная информатика» / С.В. Зыков: Интернет-университет информационных технологий. - Москва : Интернет-Университет информационных технологий, 2004. – 393 с.

7.2. Дополнительная учебная литература

1. Фаронов В.В. Delphi 6 : учебный курс / В.В. Фаронов. - Москва: Молгачева С.В. – 2003. – 664 с.: ил.
2. Семакин И.Г. Основы программирования : учебник для учреждений среднего профессионального образования / И.Г. Семакин. А.П. Шестаков. – Москва : Мастерство. 2002. – 432 с.

8. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

При проведении практических занятий используются слайдовые презентации.

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Компьютерный класс, оснащенный современными персональными компьютерами.

10. ДЛЯ СТУДЕНТОВ, ОБУЧАЮЩИХСЯ С ИСПОЛЬЗОВАНИЕМ ДИСТАНЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

При использовании электронного обучения и дистанционных образовательных технологий (далее ЭО и ДОТ) занятия полностью или частично проводятся в режиме онлайн. Объем дисциплины и распределение нагрузки по видам работ соответствует п. 4.1. Распределение баллов соответствует п. 6.2 либо может быть изменено в соответствии с решением кафедры, в случае перехода на ЭО и ДОТ в процессе обучения. Решение кафедры об используемых технологиях и системе оценивания достижений обучающихся принимается с учетом мнения ведущего преподавателя и доводится до сведения обучающихся.

Аннотация к рабочей программе дисциплины
«Объектно-ориентированное программирование»

образовательной программы высшего образования –
программы специалитета
01.05.01 – Фундаментальные математика и механика

Направленность:

Математическое и компьютерное моделирование механических систем

Трудоемкость дисциплины: 3 ЗЕ (108 академических часа)

Семестр: 7 (очная форма обучения)

Форма промежуточной аттестации: зачет.

Содержание дисциплины

Общие сведения о программах, лексемах и алфавите. Основные управляющие конструкции языка программирования C++. Массивы в C++. Функции в C++. Строки в C++. Работа с файлами в C++. Структуры в C++. Объединения в C++.