

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МУРМАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Методические указания к выполнению курсовой работы

Б1.О.09.02 Объектно-ориентированное программирование

(код и наименование дисциплины)

для направления подготовки

09.03.01 Информатика и вычислительная техника

(код и наименование направления подготовки / специальности)

направленности/профиля

Программное обеспечение вычислительной техники
и автоматизированных систем

(наименование направленности (профиля) образовательной программы)

Кафедра-разработчик:

цифровых технологий, математики и экономики

(наименование кафедры-разработчика рабочей программы)

Мурманск

2021

Составитель - Лясникова Светлана Михайловна, доцент кафедры ЦТМиЭ

Методические указания по освоению дисциплины рассмотрены и одобрены на заседании кафедры-разработчика

цифровых технологий, математики и экономики

название кафедры

21.06.2021

протокол №

12

дата

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1. СТРУКТУРА И СОДЕРЖАНИЕ КУРСОВОЙ РАБОТЫ	5
2. ТЕМЫ КУРСОВЫХ РАБОТ	6
3. ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ КУРСОВОЙ РАБОТЫ	7
4. ПОРЯДОК ЗАЩИТЫ КУРСОВОЙ РАБОТЫ.....	8
Литература	8

ВВЕДЕНИЕ

Целью дисциплины «Объектно-ориентированное программирование» является изучение основ классической теории объектно-ориентированного программирования: инкапсуляция, полиморфизм, абстракция, типизация, наследование, модульность, иерархия, понятие класса, объекта, отношения между классами и др.

В результате изучения дисциплины «Объектно-ориентированное программирование» бакалавр должен:

знать основы технологии объектно-ориентированного программирования; особенности построения объектно-ориентированных программных средств.

уметь реализовать принципы ООП при разработке программных средств; создавать диаграммы классов.

владеть приемами объектно-ориентированного решения различных задач.

В качестве базового языка используется язык C++.

Лекционный курс включает основы объектно-ориентированного программирования.

На лабораторных и практических занятиях студенты осваивают принципы объектно-ориентированного программирования на базовом языке. После выполнения курсовой работы студенты должны знать и уметь применять: основные принципы объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование.

Курсовая работа представляет собой самостоятельное решение студентом частной задачи произвольной области или проведение исследования по одному из вопросов в области программирования.

Цель выполнения курсовой работы – закрепление теоретического материала и выработка практических навыков по созданию программных средств, используя методы объектно-ориентированного программирования.

Студенту ставятся задачи по изучению теоретических источников для анализа предметной области, разбиение предметной области на объекты,

создание классов для программы, создание диаграммы классов для отображения отношения между классами, написание кода.

Данная работа предполагает активный поиск новых источников информации, моделирование практического приложения теории и разработку на этой основе эффективной схемы работы системы.

Курсовая работа выполняется студентом под контролем преподавателя. Преподаватель осуществляет консультирование по отдельным вопросам курсовой работы.

Студенту необходимо выбрать тему курсовой работы, согласовав тему с преподавателем.

Курсовая работа выносится на защиту.

1. СТРУКТУРА И СОДЕРЖАНИЕ КУРСОВОЙ РАБОТЫ

Пояснительная записка к курсовой работе должна содержать: введение, основную часть (главы и параграфы), заключение, список литературы и приложения.

Во *введение* необходимо представить:

- 1) цель работы;
- 2) актуальность работы;
- 3) описание предметной области, для которой разрабатывается программный продукт.

В *основной части* необходимо представить:

- 1) постановку задачи, которую необходимо решить;
- 2) анализ и выбор средств разработки программного продукта;
- 3) описание входных и выходных данных для всей программы;
- 4) описание алгоритмов, используемых в работе;
- 5) проектирование классов (обосновать выбор свойств и поведения для каждого класса); описание принципов объектно-

ориентированного программирования, которые необходимо использовать, оценка целесообразности этих принципов;

- б) диаграмма классов;
- 7) описание обработанных исключительных ситуаций;
- 8) описание интерфейса программного продукта;
- 9) тестирование программного продукта с тестовыми наборами данных;
- 10) краткое руководство пользователя.

В *заключении* необходимо представить основные выводы о проделанной работе, в соответствии с поставленной целью и задачей.

Список литературы должен содержать не менее 5 источников. В тексте работы необходимы ссылки на приведенные источники.

В *приложения* включают вспомогательный материал, например: скриншоты работы программы, схему работы системы, листинги всех программных модулей с подробными комментариями и др.

2. ТЕМЫ КУРСОВЫХ РАБОТ

1. Умные указатели.
2. «Годограф».
3. Геометрические фигуры на плоскости.
4. Программное средство для решения систем линейных уравнений различными методами (матричный метод, метод Гаусса, метод Жордана-Гаусса, метод Крамера).
5. Программное средство для численного интегрирования определенных интегралов различными методами (метод прямоугольников, метод трапеций, метод Монте-Карло, метод Симпсона).
6. Клиент-серверное приложение «Чат».
7. Построение множества точек на плоскости по заданной системе ограничений.

8. Шифрование текста различными алгоритмами (например, TEA, MD5 и др.).
9. Электронная записная книжка.
10. Программное средство для создания тестов, прохождения тестов и подсчета индивидуальных результатов тестирования.
11. Матрицы и векторы с элементами рациональными числами.
12. Построение графиков функций, задаваемых пользователем.
13. Нахождение производной таблично заданной функции численными методами.
14. Рациональные числа, комплексные числа и действия над ними.
15. Логические игры (например, морской бой и др.).
16. Электронный словарь.
17. Функциональный калькулятор.

Предложенные темы уточняются студентом при беседе с преподавателем.

3. ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ КУРСОВОЙ РАБОТЫ

Курсовая работа должна иметь объем 25-35 и включать в себя следующую структуру:

- 1) титульный лист;
- 2) оглавление;
- 3) введение;
- 4) основной текст (главы, параграфы);
- 5) заключение;
- 6) список использованных источников и литературы;
- 7) приложения.

При оформлении документа необходимо соблюдать следующие требования к тексту:

- 1) шрифт Times New Roman 14 стиль;

2) выравнивание – по ширине.

Страницы курсовой работы следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту, включая приложения. Номер страницы проставляют по центру сверху.

Титульный лист включают в общую нумерацию страниц документа. Номер страницы на титульном листе не ставить.

Основную часть документа следует делить на разделы (главы, соответствующие основным разделам плана работы над курсовой работой), подразделы и пункты. Разделы должны иметь порядковые номера в пределах всего документа, за исключением приложений, например: 1, 2, 3 и т.д.

4. ПОРЯДОК ЗАЩИТЫ КУРСОВОЙ РАБОТЫ

Студент допускается к защите курсовой работы при соблюдении следующих условий:

- 1) программное средство, разработанное студентом, корректно работает;
- 2) программное средство разработано с использованием ООП подхода;
- 3) пояснительная записка выполнена в соответствии п.2 методических указаний к выполнению курсовой работы.

Защита курсовых работ проводится в виде публичного выступления студента. Курсовая работа может быть оценена на «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Критерии оценки курсовой работы:

- актуальность и степень разработанности темы;
- уровень овладения теоретическим материалом;
- обоснование использованных подходов;
- соблюдение требований к оформлению курсовой работы и сроков ее исполнения.

Разработанные в ходе курсовой работы программы могут быть написаны на любом языке программирования, допускающем корректное выполнение задания.

Тексты программ должны быть снабжены подробными комментариями.

Литература

1. Гамма Э., Джонсон Р., Хелм Р., Влиссидес Д. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – М.: Питер, 2010. – 368 с.
2. Джосьютес Н. С++. Стандартная библиотека. Для профессионалов. – СПб: Питер, 2004. – 730 с.
3. Новиков Ф. А. Учебно-методическое пособие по дисциплине «Анализ и проектирование на UML». – СПб: Санкт-Петербургский университет информационных технологий, механики и оптики, 2007. – 286 с.
4. Лафоре Р. Объектно-ориентированное программирование в С++. – М.: Питер, 2004. – 922с.
5. Прата Стивен Язык программирования С++. Лекции и упражнения, 5-е изд.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2007. – 1184 с.
6. Страуструп Б. Программирование. Принципы и практика использования С++. – М.: ООО «И. Д. Вильямс», 2011. – 1248 с.
7. Скотт Мейерс Эффективное использование STL. – Питер, 2002.
8. Фаулер М. Рефакторинг. Улучшение существующего кода. – Пер. с англ. – СПб: Символ-Плюс, 2003. – 432 с.