

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДЕНО:
Директор
А. В. Замятин

Оценочные материалы по дисциплине

Объектно-ориентированное программирование

по направлению подготовки

02.03.03 Математическое обеспечение и администрирование информационных систем

Направленность (профиль) подготовки:

DevOps-инженерия в администрировании инфраструктуры ИТ-разработки

Форма обучения

Очная

Квалификация

Бакалавр

Год приема

2024

СОГЛАСОВАНО:
Руководитель ОП
С.П. Сущенко

Председатель УМК
С.П. Сущенко

Томск – 2024

1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами

ОПК-4 Способен участвовать в разработке технической документации программных продуктов и программных комплексов.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК-4.1 Обладает необходимыми знаниями нормативной базы профессиональной деятельности

ИОПК-4.2 Применяет знания нормативной базы в профессиональной деятельности

ИОПК-4.3 Разрабатывает техническую документацию, связанную с профессиональной деятельностью

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- лабораторные работы;
- устный опрос при сдаче лабораторных работ.

При сдаче каждой лабораторной работы и устном опросе проверяются знания и умения по индикаторам всех компетенций дисциплины.

Каждая лабораторная работа заключается в разработке программы по заданной теме.

Примерные темы лабораторных работ:

- Простой класс, перегрузка методов и операторов
- Агрегация, класс-контейнер
- Шаблонный класс
- Шаблонный класс-контейнер, композиция
- Наследование и виртуальные функции.

Критерии оценивания лабораторных работ

Работы оцениваются по 5 основным параметрам:

1. Полнота и правильность решения (сделано все, что и как нужно, тесты проходят)
2. Способность внести простые изменения в программу
3. Знание основных понятий и средств объектно-ориентированного программирования на C++
4. Понимание назначения используемых методов (функций) и умение правильно их реализовать
5. Ориентация в программе, основных переменных/атрибутах, функциях/методах.

Работы оцениваются в баллах – 15 баллов за каждую (максимально 75 баллов за 5 работ).

Для получения итоговой оценки по курсу нужно сдать не менее четырех заданий.

3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания

Зачет с оценкой в третьем семестре проводится в форме коллоквиума в конце семестра по всем темам курса.

На коллоквиуме студент получает по 5 вопросов и отвечает устно. За каждый ответ можно получить от 0 до 5 баллов (максимально 25 баллов за коллоквиум).

Для получения итоговой оценки по курсу нужно набрать на коллоквиуме не менее пяти баллов.

При ответах на вопросы на коллоквиуме проверяются знания и умения по индикаторам всех компетенций дисциплины.

При формировании итоговой оценки суммируются баллы за лабораторные задания и баллы за коллоквиум. Итоговая оценка:

80 – 100 баллов - отлично

60 – 79 баллов – хорошо

40 – 59 баллов – удовлетворительно

меньше 40 баллов – неудовлетворительно.

4. Оценочные материалы для проверки остаточных знаний (сформированности компетенций)

Список вопросов для коллоквиума (около 30 вопросов) можно использовать и при проверке остаточных знаний.

Список вопросов для оценки остаточных знаний

1. Основные принципы ООП
2. Понятие и структура класса
3. Основные принципы – инкапсуляция, полиморфизм, наследование
4. Объекты – создание и разрушение
5. Конструкторы с параметрами и без
6. Конструкторы копий
7. Деструкторы
8. Перегрузка функций
9. Указатель `this`
10. Перегрузка бинарных операторов
11. Перегрузка унарных операторов
12. Дружественные функции и операторы
13. Ссылки
14. Принципы и модификаторы наследования
15. Конструкторы и деструкторы производных классов
16. Множественное наследование
17. Иерархии классов, ссылки и указатели на производные типы
18. Виртуальные функции
19. Абстрактные классы
20. Статический и динамический полиморфизм
21. Обработка исключений
22. Генерация и перехват исключений
23. Работа с динамической памятью в C++
24. Система ввода-вывода в C++
25. Потоки ввода-вывода
26. Перегрузка оператора вставки
27. Перегрузка оператора извлечения
28. Шаблонные функции
29. Шаблонные классы
30. Структура библиотеки STL
31. Основные типы контейнеров STL.

Информация о разработчиках

Фукс Александр Львович, канд. техн. наук, доцент кафедры теоретических основ информатики ТГУ