

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ<sup>1</sup>**  
**по дисциплине**

**« Технологии и методы программирования »**

Направление подготовки	<i>01.03.04 Прикладная математика</i>
Направленность (профиль) образовательной программы	<i>Математическое и компьютерное моделирование</i>

Обеспечивающее подразделение
<i>Кафедра «Проектирование, управление и разработка информационных систем»</i>

Разработчик ФОС:

Старший преподаватель \_\_\_\_\_ Шамак В.А.  
(должность, степень, ученое звание) (подпись) (ФИО)

Оценочные материалы по дисциплине рассмотрены и одобрены на заседании  
кафедры, протокол № \_\_\_\_\_ от « \_\_\_\_ » \_\_\_\_\_ 2023 г.

Заведующий кафедрой \_\_\_\_\_ Петрова А.Н.  
ПУРИС (наименование кафедры) (подпись) (ФИО)

<sup>1</sup> В данном приложении представлены типовые оценочные средства. Полный комплект оценочных средств, включающий все варианты заданий (тестов, контрольных работ и др.), предлагаемых обучающемуся, хранится на кафедре в бумажном и электронном виде.

## 1 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами образовательной программы

Таблица 1 – Компетенции и индикаторы их достижения

Код и наименование компетенции	Индикаторы достижения	Планируемые результаты обучения по дисциплине
<b>Общепрофессиональные</b>		
ОПК-4 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-4.1 Знает основные принципы составления алгоритмов и программ; ОПК-4.2 Умеет разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения; ОПК-4.3 Владеет навыками разработки алгоритмов и компьютерных программ, пригодных для практического применения	Знает алгоритмические языки программирования, современные среды разработки программного обеспечения.  Умеет составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы.  Владеет навыками программирования, отладки и тестирования программ

Таблица 2 – Паспорт фонда оценочных средств

Контролируемые разделы (темы) дисциплины	Формируемая компетенция	Наименование оценочного средства	Показатели оценки
очная форма обучения			
<b>Тема 1. Классы</b>	ОПК-4	Задание 1. Классы, РГР, Экзамен	Владеет навыками разработки ПО на базе классов
<b>Тема 2. Наследование</b>	ОПК-4	Задание 2. Наследование, РГР, Экзамен	Владеет навыками разработки ПО на базе объектно-ориентированного программирования
<b>Тема 3. Шаблоны классов</b>	ОПК-4	Задание 3. Шаблоны классов и обработка исключительных ситуаций, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов

<b>Контролируемые разделы (темы) дисциплины</b>	<b>Формируемая компетенция</b>	<b>Наименование оценочного средства</b>	<b>Показатели оценки</b>
<b>Тема 4. Обработка исключительных ситуаций</b>	ОПК-4	Задание 4. Обработка исключительных ситуаций, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 5. Преобразования типов</b>	ОПК-4	Задание 5. Преобразования типов, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 6. Потокосы классы</b>	ОПК-4	Задание 6. Потокосы классы, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 7. Строки и регулярные выражения</b>	ОПК-4	Задание 7. Строки и регулярные выражения, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 8. Контейнерные классы</b>	ОПК-4	Задание 8. Контейнерные классы, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 9. Итераторы и функторы</b>	ОПК-4	Задание 9. Итераторы и функторы, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 10. Алгоритмы</b>	ОПК-4	Задание 10. Алгоритмы, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 11. Средства для численных расчетов</b>	ОПК-4	Задание 11. Средства для численных расчетов, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 12. Стандарты языка и сторонние библиотеки</b>	ОПК-4	Задание 12. Стандарты языка и сторонние библиотеки, РГР, Экзамен	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения

## 2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, представлены в виде технологической карты дисциплины (таблица 3).

Таблица 3 – Технологическая карта

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
3 семестр <i>Промежуточная аттестация в форме экзамена</i>				
1	Задание 1 «Классы»	4 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
2	Задание 2 «Наследование»	6 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
3	Задание 3. «Шаблоны классов и обработка и исключительных ситуаций»	8 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
4	Задание 4 «Потоки и контейнеры»	10 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
5	Задание 5 «Строки и регулярные выражения»	12 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
6	Задание 6	14	5	0 баллов – задание не выполнено или выполнено не верно

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
	«Контейнеры и алгоритмы»	неделя		3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
7	Задание 7 «Класс valarray»	15 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
8	Расчетно-графическая работа (7 разделов)	15 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
9	Контрольная работа	17 неделя	35	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
Текущий контроль:		-	70 баллов	-
Экзамен:		-	30 баллов	-
ИТОГО:		-	100 баллов	-
<b>Критерии оценки результатов обучения по дисциплине:</b> 0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для промежуточной аттестации по дисциплине); 65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень); 75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)				

- 3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций в ходе освоения образовательной программы**

**Задания для текущего контроля успеваемости**

## Комплект заданий для текущей аттестации (семестр 3)

**Задание 1.** Описать класс в нотации UML и разработать программу, демонстрирующую все возможности класса. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса.

### 1. Вариант 1

Описать класс, реализующий стек. Написать программу, использующую этот класс для моделирования T-образного сортировочного узла на железной дороге.

Программа должна разделять на два направления состав, состоящий из вагонов двух типов (на каждое направление формируется состав из вагонов одного типа). Предусмотреть возможность формирования состава из файла и с клавиатуры.

### 2. Вариант 2

Описать класс, реализующий бинарное дерево, обладающее возможностями добавления новых элементов, удаления существующих, поиска элемента по ключу, а также последовательного доступа ко всем элементам.

Написать программу, использующую этот класс для представления англо-русского словаря. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса. Предусмотреть возможность создания словаря из файла и с клавиатуры.

**Задание 2.** Реализовать абстрактный базовый класс и производные от него. Во всех классах должно быть хотя бы одно поле, память под которое выделяется динамически. Используя механизм виртуальных методов, продемонстрировать работу с производным классом через указатель на базовый класс. Обеспечить удобные средства получения доступа к состоянию класса. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса.

### 1. Вариант 1

Создать абстрактный класс File, инкапсулирующий в себе методы Open, Close, Seek, Read, Write, GetPosition и GetLength. Создать производные классы MyDataFile и MyDataFile2 — файлы, содержащие в себе данные некоторого определенного типа MyData1 и MyData2, а также заголовки, облегчающие доступ к этим файлам.

Создать класс Folder, содержащий список (или массив) объектов этих классов в динамической памяти. Предусмотреть возможность вывода списка имен и длин файлов. Написать демонстрационную программу, в которой будут использоваться все методы классов.

### 2. Вариант 2

Создать абстрактный класс Point (точка). На его основе создать классы Colored Point и Line. На основе класса Line создать классы Colored Line и Poly Line (многоугольник). Все классы должны иметь виртуальные методы установки и получения значений всех координат, а также изменения цвета и получения текущего цвета.

Создать класс Picture, содержащий список (или массив) объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

**Задание 3.** Реализовать шаблонный класс. Программа должна содержать меню, позволяющее выполнить проверку работоспособности шаблона класса как для встроенных, так и для пользовательских типов данных. Предусмотреть обработку ошибок с помощью механизма исключительных ситуаций.

### 1. Вариант 1

Создать шаблон класса Stack. Использовать его при решении задачи варианта 1 задания 1.

### 2. Вариант 2

Создать шаблон класса BinaryTree. Использовать его при решении задачи варианта 2 задания 1.

**Задание 4.** Выполнить задание 2, используя файловые и стандартные потоки, а также контейнеры. Виды контейнеров выбрать исходя из потребностей задачи. Во всех вариантах использовать работу с файлами и обработку ошибок.

**Задание 5.** Выполнить задание с использованием класса string и регулярных выражений.

1. Вариант 1

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.

2. Вариант 2

Написать программу, которая считывает текст из файла и выводит на экран только предложения, содержащие введенное с клавиатуры слово.

**Задание 6.** Выполнить задание с использованием подходящих контейнерных классов и алгоритмов библиотеки. Содержимое контейнеров (не менее миллиона элементов) формировать с помощью генераторов случайных чисел.

1. Вариант 1

Найти сумму отрицательных элементов массива.

Найти произведение элементов массива, расположенных между максимальным и минимальным элементами.

Упорядочить элементы массива по возрастанию.

2. Вариант 2

Найти сумму положительных элементов массива.

Найти произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.

Упорядочить элементы массива по убыванию.

**Задание 7.** Выполнить задания с использованием класса valarray, срезов и генераторов случайных чисел.

1. Вариант 1

Дана целочисленная прямоугольная матрица. Определить:

- количество строк, не содержащих ни одного нулевого элемента;

- максимальное из чисел, встречающихся в заданной матрице более одного раза.

2. Вариант 2.

Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.

Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.

### **Расчетно-графическая работа**

1. Выполнить верификацию и тестирование кода, полученного в ходе текущей аттестации при выполнении индивидуальных заданий, путем проведения тестовых запусков программ на заранее подготовленных наборах данных.

2. Описать решения индивидуальных заданий полученные в ходе прохождения те- кущей аттестации в соответствии с требованиями единой системы программной документации (ЕСПД) и РД 013-2016 «Текстовые студенческие

работы. Правила оформления».

Структурными элементами данной работы должны быть:

- титульный лист;
- текст всех заданий (в соответствии с вариантом);
- содержание;
- введение
- основная часть;
- заключение;
- список использованных источников;
- приложения.

Во введении дается краткое описание изучаемой дисциплины, которой посвящена данная работа, а также приводится обзор выполненной работы.

Каждое выполненное индивидуальное задание оформляется отдельным разделом основной части отчета. Каждый раздел включает теоретические сведения по теме разделас обязательным указанием ссылок на источники информации, в случае её заимствования. Обязательными подразделами каждого отчета являются «Описание программы», «Текст программы», «Программа и методика испытаний». Требования к содержанию подразделовприведены в таблице:

<b>Вид программного документа</b>	<b>Содержание программного документа</b>
Описание программы	Сведения о логической структуре и функционировании программы
Текст программы	Запись программы с необходимыми комментариями
Программа и методика испытаний	Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля

В заключении приводятся общие выводы и предложения, а также краткое описание проделанной работы; даются практические рекомендации. (1,5 - 2 страницы).

Список литературы состоит из нормативно-правовых актов, учебников и учебных пособий, научных статей, использованных в ходе выполнения индивидуального задания.

Приложения помещают после списка литературы в порядке их отсылки или обращения к ним в тексте. В качестве приложений рекомендуется предоставлять копии документов, бланков договоров, организационно-распорядительных документов, аналитических таблиц, иных документов, иллюстрирующих содержание основной части.

Отчет предварительно оценивается и допускается к защите после проверки его соответствия требованиям, предъявляемым данными методическими указаниями. Защита отчетов организуется в форме собеседования.

### **Задания для промежуточной аттестации**

#### **Контрольные вопросы к экзамену**

1. Абстрактные структуры данных.
2. Динамические структуры данных.
3. Описание класса.
4. Описание объектов.
5. Указатель this.
6. Конструкторы объектов.



7. Статические элементы класса.
8. Дружественные функции классов.
9. Деструкторы.
10. Операции класса.
11. Указатели на элементы класса.
12. Вложенные и локальные классы.
13. Каноническая форма класса в C++11.
14. Формы наследования.
15. Формы вложения.
16. Создание шаблона классов.
17. Использование шаблона классов.
18. Специализация шаблона классов.
19. Принцип обработки исключений.
20. Генерация исключений.
21. Перехват исключений.
22. Список исключений функции.
23. Исключения в конструкторах и деструкторах.
24. Стандартные исключения.
25. Операции приведения типов в C.
26. Операции приведения типов в C++.
27. Динамическое определение типа.
28. Стандартные потоки.
29. Файловые потоки.
30. Строковые потоки.
31. Перегрузка операций ввода-вывода.
32. Строки.
33. Регулярные выражения.
34. Последовательные контейнеры.
35. Ассоциативные контейнеры.
36. Итераторы.
37. Функторы.
38. Немодифицирующие операции с последовательностями.
39. Модифицирующие операции с последовательностями.
40. Алгоритмы, связанные с сортировкой.
41. Обобщенные численные алгоритмы.
42. Класс `valarray`.

### **Примерные варианты экзаменационных билетов**

#### **Билет №1**

1. Абстрактные структуры данных.
2. Дружественные функции классов.
3. Формы вложения.
4. Список исключений функции.

#### **Билет №2**

1. Статические элементы класса.
2. Формы наследования.
3. Перехват исключений.