

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Астраханский государственный университет имени В. Н. Татищева»
(Астраханский государственный университет им. В. Н. Татищева)

СОГЛАСОВАНО
Руководитель ОПОП

УТВЕРЖДАЮ
Зав. кафедрой ПМИ

_____ М.В. Коломина

_____ М.В. Коломина

«5» апреля 2024 г.

«5» апреля 2024 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ**

Составители	Корнеев Г.А., к.т.н., доцент ФИТиП, ИТМО Смирнов А.П., к.ф.-м.н., доцент каф. ПМИ, АГУ Белов С.В., директор ООО «ТРАСТ ПОИНТ» Измайлов Г.А., генеральный директор ООО «Агент Плюс»
Согласовано с работодателями	01.03.02 Прикладная математика и информатика
Направление подготовки / специальность	Программирование и искусственный интеллект
Направленность (профиль) ОПОП	бакалавр
Квалификация (степень)	очная
Форма обучения	2024
Год приёма	1,2
Курс	2,3
Семестр(ы)	

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

1.1. Целями освоения дисциплины (модуля) «Практикум по программированию» являются:

- ознакомление студентов с основными понятиями и принципами при разработке программ на языке программирования C++;
- ознакомление со средами программирования, с теоретическими достижениями в этой области.

1.2. Задачи освоения дисциплины: углубление знаний в области программирования на языке C++, развитие практических навыков в области прикладной математики и информатики, подготовка к изучению других дисциплин.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

2.1. Учебная дисциплина «Практикум по программированию» относится к части, формируемая участниками образовательных отношений обязательной части и осваивается в 2,3 семестрах.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс освоения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО и ОПОП ВО по данному направлению подготовки / специальности:

а) профессиональных (ПК).

- ПК-1. Способен создавать, отлаживать и оформлять программный код
- ПК-3. Способен обеспечивать заданный уровень производительности, надежности и безопасности при создании вариантов архитектуры программного средства
- ПК-4. Способен реализовывать программные средства
- ПК-5. Способен участвовать в разработке операционной системы

Таблица 1. Декомпозиция результатов обучения

Код компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине (модулю)		
		Знать (1)	Уметь (2)	Владеть (3)
ПК-1	ПК-1.1. Способен осуществлять формализацию и алгоритмизацию поставленных задач. ПК-1.2. Способен создавать программный код с использованием языков программирования, определения и манипулирования данными. ПК-1.3. Способен работать с системой контроля версий, оформлять в соответствии с требованиями, проверять и отлаживать программный код.	Ключевые приемы программирования базовых структур данных.	Проектировать программный продукт на базовом уровне.	Навыками использования отладчика.
ПК-3	ПК-3.3. Способен определять перечень возможных моделей обеспечения отказоустойчивости программных компонентов, осуществлять проектную оценку надежности компонентов программного средства. ПК-3.4. Способен определять перечень возможных технологий доступа к данным, выбирать протоколы взаимодействия компонентов, определять алгоритмы компонентов, включая методы и схемы ПК-3.5. Способен осуществлять оценку и выбор стиля написания кода, моделей управления исключениями, управления и мониторинга критически важных событий	Низкоуровневые особенности функционирования программ.	Использовать стандартные библиотеки.	Основными приемами программирования.

ПК-4	ПК-4.1. Способен анализировать качество кода. ПК-4.2. Способен проводить испытания программного средства и его компонентов	Устройство курса языка программирования C++, последовательность представления.	Составлять задачи для курса языка программирования C++ с автоматической проверкой.	Навыками поиска ошибок в коротких программах, решающих задачи курса языка программирования C++.
ПК-5	ПК-5.1. Способен формировать требования к операционной системе	Архитектура механизма исключений, шаблонов.	Разрабатывать программы на языке c++ в парадигме ООП	Навыками отладки программы на языке c++.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины в соответствии с учебным планом составляет 2 зачетные единицы (72 часа).

Трудоемкость отдельных видов учебной работы студентов очной формы обучения приведена в таблице 2.1.

Таблица 2.1. Трудоемкость отдельных видов учебной работы по формам обучения

Вид учебной и внеучебной работы	для очной формы обучения
Объем дисциплины в зачетных единицах	2
Объем дисциплины в академических часах	72
Контактная работа обучающихся с преподавателем (всего), в том числе (час.):	72
- занятия лекционного типа, в том числе:	0
- практическая подготовка (если предусмотрена)	0
- занятия семинарского типа (семинары, практические, лабораторные), в том числе:	72
- практическая подготовка (если предусмотрена)	0
- консультация (предэкзаменационная)	0
- промежуточная аттестация по дисциплине	0
Самостоятельная работа обучающихся (час.)	0
Форма промежуточной аттестации обучающегося (зачет/экзамен), семестр (ы)	Зачет, 2-3 семестр

Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий и самостоятельной работы представлены в таблице 2.2.

Таблица 2.2. Структура и содержание дисциплины (модуля) (отсутствует ИТОГО часов)

Раздел, тема дисциплины (модуля)	Контактная работа, час.							СР, час.	Итого ча- сов	Форма текущего контроля успеваемости, форма промежуточной аттестации <i>[по семестрам]</i>
	Л		ПЗ		ЛР		КР / КП			
	Л	в т.ч. ПП	ПЗ	в т.ч. ПП	ЛР	в т.ч. ПП				
Семестр 2										
СРР и Ассемблер					36				36	Лабораторная работа №1-2, Контрольная работа №1
ИТОГО за семестр					36				36	
Семестр 3										
Память					36				36	Лабораторная работа №3-4, Контрольная работа №2
ИТОГО за семестр					36				36	
Консультации										
Контроль промежуточной аттестации										Зачет
ИТОГО за весь период					72				72	

Примечание: Л – лекция; ПЗ – практическое занятие, семинар; ЛР – лабораторная работа; ПП – практическая подготовка; КР / КП – курсовая работа / курсовой проект; КПА – контроль промежуточной аттестации; КС – консультации; СР – самостоятельная работа

Таблица 3. Матрица соотнесения разделов, тем учебной дисциплины (модуля) и формируемых компетенций

Раздел, тема дисциплины (модуля)	Кол-во часов	Код компетенции				Общее количество компетенций
		ПК-1	ПК-3	ПК-4	ПК-5	
СРР и Ассемблер	36	+	+	+	+	4
Память	36	+	+	+	+	4

Краткое содержание каждой темы дисциплины (модуля)

№ раз-дела	Наименование раздела дисциплины	Содержание
1	СРР и Ассемблер	ОПП. Линковка, Assembler. Регистры. Многозадачность. Кеш-память, C-C++. Стандартная библиотека. Компиляция
2	Память	STL. Пространства имен. Неопределенное поведение. Указатели. Перегрузки указателей. Анонимные функции. Boost. Многопоточность, RAII. Аллокация памяти. Шаблоны

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРЕПОДАВАНИЮ И ОСВОЕНИЮ ДИСЦИПЛИНЫ**5.1. Указания для преподавателей по организации и проведению учебных занятий по дисциплине (модулю)****Лабораторные занятия**

Лабораторное занятие – целенаправленная форма организации педагогического процесса, направленная на углубление научно-теоретических знаний и овладение определенными методами работы, в процессе которых вырабатываются умения и навыки выполнения тех или иных учебных действий в данной сфере науки. Они развивают научное мышление и речь, позволяют проверить знания студентов и выступают как средства оперативной обратной связи.

Правильно организованные лабораторные занятия ориентированы на решение следующих задач:

- обобщение, систематизация, углубление, закрепление полученных на лекциях и в процессе самостоятельной работы теоретических знаний по дисциплине (предмету);
- формирование практических умений и навыков, необходимых в будущей профессиональной деятельности, реализация единства интеллектуальной и практической деятельности;
- выработка при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

Состав заданий для лабораторного занятия должен быть спланирован с расчетом, чтобы за отведенное время они могли быть качественно выполнены большинством учащихся.

Лабораторные занятия должны так быть организованы, чтобы студенты ощущали нарастание сложности выполнения заданий, испытывали бы положительные эмоции от переживания собственного успеха в учении, поисками правильных и точных решений.

5.2. Указания для обучающихся по освоению дисциплины (модулю)**Лабораторное занятие**

- Лабораторное занятие – наиболее активный вид учебных занятий в вузе. Он предполагает самостоятельную работу над лекциями и учебными пособиями.
- К каждому лабораторному занятию нужно готовиться. Подготовку следует начинать с повторения теории (по записям лекций или по учебному пособию). После этого нужно решать задачи из предложенного домашнего задания.

5.3. Виды и формы письменных работ, предусмотренных при освоении дисциплины (модуля), выполняемые обучающимися самостоятельно

В рамках дисциплины не предусмотрено

6. ОБРАЗОВАТЕЛЬНЫЕ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

При реализации различных видов учебной работы по дисциплине «Практикум по программированию» могут использоваться электронное обучение и дистанционные образовательные технологии.

6.1. Образовательные технологии

При реализации различных видов учебной работы по дисциплине «Практикум по программированию» могут использоваться электронное обучение и дистанционные образовательные технологии.

Таблица 5. Образовательные технологии, используемые при реализации учебных занятий

Раздел, тема дисциплины	Форма учебного занятия		
	Лекция	Практическое занятие, семинар	Лабораторная работа
CPP и Ассемблер	<i>Не предусмотрено</i>	<i>Не предусмотрено</i>	<i>Выполнение лабораторной работы</i>
Память	<i>Не предусмотрено</i>	<i>Не предусмотрено</i>	<i>Выполнение лабораторной работы</i>

6.2. Информационные технологии

При реализации различных видов учебной и внеучебной работы используются следующие информационные технологии:

- система управления обучением LMS Moodle;
- использование возможностей Интернета в учебном процессе (рассылка заданий, предоставление выполненных работ, ответы на вопросы, ознакомление обучающихся с оценками и т.д.);
- использование электронных учебников и различных сайтов (например, электронные библиотеки, журналы и т.д.) как источник информации;
- использование возможностей электронной почты;
- использование средств представления учебной информации (электронных учебных пособий, применение новых технологий для проведения занятий с использованием презентаций и т.д.);
- использование интерактивных средств взаимодействия участников образовательного процесса (технологии дистанционного или открытого обучения в глобальной сети);
- использование интегрированных образовательных сред, где главной составляющей являются не только применяемые технологии, но и содержательная часть, т.е. информационные ресурсы (доступ к мировым информационным ресурсам, на базе которых строится учебный процесс).

6.3. Программное обеспечение, современные профессиональные базы данных и информационные справочные системы

6.3.1. Программное обеспечение

Перечень программного обеспечения (*состав подлежит обновлению при необходимости*)

Наименование программного обеспечения	Назначение
Adobe Reader	Программа для просмотра электронных документов
LMS Moodle	Образовательный портал ФГБОУ ВО «АГУ»
Microsoft Office	Пакет офисных программ
OpenOffice	Пакет офисных программ
7-zip	Архиватор
Microsoft Windows	Операционная система
Kaspersky Endpoint Security	Средство антивирусной защиты
Google Chrome	Браузер
Opera	Браузер
Microsoft Visual Studio	Среда разработки
VirtualBox	Программный продукт виртуализации операционных систем

CodeBlocks	Кроссплатформенная среда разработки
WinDjView	Программа для просмотра файлов в формате DJV и DjVu

6.3.2. Современные профессиональные базы данных и информационные справочные системы

1. Электронная библиотека «Астраханский государственный университет» собственной генерации на платформе ЭБС «Электронный Читальный зал – БиблиоТех». <https://biblio.asu.edu.ru>.

2. Электронно-библиотечная система (ЭБС) ООО «Политехресурс» «Консультант студента». <https://www.studentlibrary.ru>.

3. Электронный каталог Научной библиотеки АГУ на базе MARK SQL НПО «Информ-систем». <https://library.asu.edu.ru>.

7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

7.1. Паспорт фонда оценочных средств

При проведении текущего контроля и промежуточной аттестации по дисциплине «Практикум по программированию» проверяется сформированность у обучающихся компетенций, указанных в разделе 3 настоящей программы. Этапность формирования данных компетенций в процессе освоения образовательной программы определяется последовательным освоением дисциплин и прохождением практик, а в процессе освоения дисциплины – последовательным достижением результатов освоения содержательно связанных между собой разделов, тем.

Таблица 6. Соответствие разделов, тем дисциплины, результатов обучения по дисциплине и оценочных средств

№ п/п	Контролируемые разделы, темы дисциплины (модуля)	Код контролируемой компетенции (компетенций)	Наименование оценочного средства
1	СРР и Ассемблер	ПК-1, ПК-3, ПК-4, ПК-5	лабораторная работа
2	Память	ПК-1, ПК-3, ПК-4, ПК-5	лабораторная работа

7.2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Таблица 7. Показатели оценивания результатов обучения в виде знаний

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует глубокое знание теоретического материала, умение обоснованно излагать свои мысли по обсуждаемым вопросам, способность полно, правильно и аргументированно отвечать на вопросы, приводить примеры
4 «хорошо»	демонстрирует знание теоретического материала, его последовательное изложение, способность приводить примеры, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует неполное, фрагментарное знание теоретического материала, требующее наводящих вопросов преподавателя, допускает существенные ошибки в его изложении, затрудняется в приведении примеров и формулировке выводов
2 «неудовлетворительно»	демонстрирует существенные пробелы в знании теоретического материала, не способен его изложить и ответить на наводящие вопросы преподавателя, не может привести примеры

Таблица 8. Показатели оценивания результатов обучения в виде умений и владений

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы
4 «хорошо»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует отдельные, несистематизированные навыки, испытывает затруднения и допускает ошибки при выполнении заданий, выполняет задание по подсказке преподавателя, затрудняется в формулировке выводов
2 «неудовлетворительно»	не способен правильно выполнить задания

7.3. Контрольные задания и иные материалы, необходимые для оценки результатов обучения по дисциплине (модулю)

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторная работа 1

Лабораторная работа выполняется в рамках каждого раздела курса с целью усвоения прослушанного студентом теоретического материала.

Объем выполненной работы: каждая лабораторная работа содержит 3-5 задач.

Срок сдачи работы: лабораторные работы должны быть сданы в период прочтения курса. Сдача работы представляет собой предоставление отчёта в свободной форме в письменном или электронном виде и, в случае необходимости, устные ответы на уточняющие вопросы по отдельным задачам.

Пример заданий для лабораторной работы 1

1. Необходимо написать класс длинное знаковое число.

Требования к решению следующие: Реализация должна содержаться в классе `big_integer` и находиться в файле `big_integer.h`.

Класс должен содержать: Конструктор по умолчанию, инициализирующий число нулем. Конструктор копирования, после которого исходный объект и копию можно изменять независимо. Конструктор от `int`. Explicit конструктор от `std::string`. Оператор присваивания. Операторы сравнения.

Арифметические операции: сложение, вычитание, изменение знака (унарный минус), унарный плюс. Умножение работающее за время $O(nm)$, где n, m — длины множителей в битах. Деление и остаток от деления работающие за время $O(nm)$, где n — длина делителя в битах, а m — длина частного в битах. Префиксный/постфиксный инкремент/декремент. (опционально)

Битовые операции: и, или, исключающее или, не. Битовые сдвиги.

Должна существовать глобальная функция `std::string to_string(big_integer const&)` возвращающая строковое представление числа. Реализация функций класса должна быть расположена в файле `big_integer.cpp`.

Пользоваться сторонними библиотеками длинных чисел при выполнении этого задания нельзя. Разряды числа должны представляться 32-битными либо 64-битными числами. При этом необходимо, чтобы все биты в их представлении использовались. Реализация должна использовать арифметические операции той битности, которая естественна для данного представления разрядов.

2. Провести проверку корректности программы с помощью набора модульных тестов: <https://github.com/sorokin/cpp-course/tree/master/bigint>.

Отчет по лабораторной работе №_____

1. Цель и задачи лабораторной работы:
2. Текстовое описание способа реализации кода:
3. Текст кода:
4. Результаты работы программы в виде набора входных и выходных параметров:
0. Выводы:

Лабораторная работа 2

Лабораторная работа выполняется в рамках каждого раздела курса с целью усвоения прослушанного студентом теоретического материала.

Объем выполненной работы: каждая лабораторная работа содержит 3-5 задач.

Срок сдачи работы: лабораторные работы должны быть сданы в период прочтения курса. Сдача работы представляет собой предоставление отчёта в свободной форме в письменном или электронном виде и, в случае необходимости, устные ответы на уточняющие вопросы по отдельным задачам.

Пример заданий для лабораторной работы 2

1. В данном задании необходимо реализовать класс длинного числа со small-object и copy-on-write оптимизациями. Длинное число должно удовлетворять всем требованиям задания 2 плюс: Если в `big_integer` хранится число $-2N \leq a \leq 2N-1$, `big_integer` не должен выделять динамическую память. Конкретное значение N вы можете выбрать самостоятельно, допустимыми являются любые $N \geq 30$. Числа $a < -2N$ и $a > 2N-1$ должны выделять не больше одного блока динамической памяти на каждый экземпляр `big_integer`. Конструктор копирования и оператор присваивания должны работать за $O(1)$ и удовлетворять гарантии безопасности исключений `nothrow`

Отчет по лабораторной работе №_____

1. Цель и задачи лабораторной работы:
2. Текстовое описание способа реализации кода:
3. Текст кода:
4. Результаты работы программы в виде набора входных и выходных параметров:
5. Выводы:

Лабораторная работа 3

Лабораторная работа выполняется в рамках каждого раздела курса с целью усвоения прослушанного студентом теоретического материала.

Объем выполненной работы: каждая лабораторная работа содержит 3-5 задач.

Срок сдачи работы: лабораторные работы должны быть сданы в период прочтения курса. Сдача работы представляет собой предоставление отчёта в свободной форме в письменном или электронном виде и, в случае необходимости, устные ответы на уточняющие вопросы по отдельным задачам.

Пример заданий для лабораторной работы 2

1. В данном задании необходимо реализовать класс длинного числа со small-object и copy-on-write оптимизациями. Длинное число должно удовлетворять всем требованиям задания 2 плюс: Если в `big_integer` хранится число $-2N \leq a \leq 2N-1$, `big_integer` не должен выделять динамическую память. Конкретное значение N вы можете выбрать самостоятельно, допустимыми являются любые $N \geq 30$. Числа $a < -2N$ и $a > 2N-1$ должны выделять не больше одного блока динамической памяти на каждый экземпляр `big_integer`. Конструктор копирования и оператор присваивания должны работать за $O(1)$ и удовлетворять гарантии безопасности исключений `nothrow`

Отчет по лабораторной работе №_____

1. Цель и задачи лабораторной работы:
2. Текстовое описание способа реализации кода:
3. Текст кода:

4. Результаты работы программы в виде набора входных и выходных параметров:
5. Выводы:

Лабораторная работа 4

Лабораторная работа выполняется в рамках каждого раздела курса с целью усвоения прослушанного студентом теоретического материала.

Объем выполненной работы: каждая лабораторная работа содержит 3-5 задач.

Срок сдачи работы: лабораторные работы должны быть сданы в период прочтения курса. Сдача работы представляет собой предоставление отчёта в свободной форме в письменном или электронном виде и, в случае необходимости, устные ответы на уточняющие вопросы по отдельным задачам.

Пример заданий для лабораторной работы 3

1. В данном задании необходимо реализовать библиотеку для сжатия Хаффманом. Программа должна состоять из трех частей: Библиотека, которая реализует операции сжатия и распаковки. Консольная утилита, которая позволяет сжимать/распаковывать файлы на диске. Программа с набором тестов, проверяющая корректность библиотеки.

Отчет по лабораторной работе № _____

1. Цель и задачи лабораторной работы:
2. Текстовое описание способа реализации кода:
3. Текст кода:
4. Результаты работы программы в виде набора входных и выходных параметров:
5. Выводы:

Пример контрольной работы

1. Покажите, как сделать сортировку слиянием снизу вверх, без рекурсии. Использовать датасет, представленный по ссылке

<https://drive.google.com/file/d/1WsKfIHM954fRHSK0iasEH3MrpeCl4Qde/view?usp=sharing>

2. Дан массив a . Найдите число инверсий в нем. Время $O(n \log n)$. В качестве элементов массива использовать столбец D из датасета, представленного по ссылке

https://docs.google.com/spreadsheets/d/1c-E3k3_MHWIQeMfX8V4t3Cxe1EYdpIp/edit?usp=sharing&ouid=113904695302942283106&rtpof=true&sd=true

Отчет по контрольной работе № _____

1. Текстовое описание способа реализации кода:
2. Текст кода:
3. Результаты работы программы в виде набора входных и выходных параметров:

Перечень вопросов, выносимых на зачет

1. i8086
2. Регистры общего назначения, регистр IP
3. Распространенные арифметические операции: add, sub, mul, div, and, or, xor, shl, shr, sar
4. Команды безусловных переходов
5. Регистр флагов и некоторые биты в нём (ZF, CF, OF, SF)
6. Команды условных переходов
7. Команды cmp и test
8. i80386, 32-битные регистры
9. Регистр SP и команды для работы со стеком (push, pop)
10. Команды call и ret
11. Регистр BP, стековый фрейм
12. Прерывания
13. Вытесняющая многозадачность
14. Защищенный режим, страничная адресация

15. Кеш-память
16. Спекулятивное исполнение команд
17. Branch prediction
18. Пересечение языков C и C++
19. Целочисленные типы, ABI
20. Числа с плавающей точкой
21. Соглашения вызова функций
22. Указатели
23. Операции взятия адреса и разыменования
24. Массивы, неявное приведение к указателю
25. Операции инкремента, декремента, сложения с числом, вычитание числа и разности указателей
26. Синтаксис указателей и массивов в общем случае, использование скобок для указания приоритета операций
27. Указатели на функцию
28. Процесс компиляции программ
29. Объектные файлы
30. Трансляция, единица трансляция
31. Линковка
32. Объявления и определения для переменных и для функций
33. Модификатор static
34. Препроцессор, примеры директив
35. Header-файлы
36. Include guard
37. Понятие константности
38. Взаимодействие константности с указателями, указатель на константу, константный указатель
39. Модификатор inline, правило одного определения (ODR)
40. Классы
41. Функции члены класса, неявный аргумент this
42. Конструктор класса и деструктор
43. Время жизни объекта: для локальных переменных, для глобальных переменных, для временных объектов, для данных класса
44. Список инициализации конструктора
45. Ссылки
46. Конструктор копирования
47. Оператор присваивания
48. Перегрузка операторов

Таблица 9. Примеры оценочных средств с ключами правильных ответов

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
ПК-1. Способен создавать, отлаживать и оформлять программный код				
1.	Задание закрытого типа	Отметьте правильный вариант описания функции шаблона: (1) <code>template (class T) void change(T *p1, T *p2) { . . . };</code> (2) <code>template <class T>; void change<T *p1, T *p2> { . . . };</code> (3) <code>template <class T> void change(T *p1, T *p2) { . . . };</code> (4) <code>template {class T} void change(T *p1, T *p2) (. . .);</code>	3	4
2.		Программа на языке Си++ начинает выполняться с: (1) первой функции в программе (2) функции main	2	4

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		(3) той функции, которая указана как стартовая при компиляции программы		
3.		Если есть два объявления <code>int qwerty; int QWERTY;</code> какое из утверждений верно (1) такие имена переменных недопустимы (2) объявления правильные (3) такие объявления недопустимы, так как мы пытаемся создать две одинаковые переменные	2	4
4.		Операция <code>++</code> (1) увеличивает значение переменной на единицу (2) увеличивает значение переменной на два (3) уменьшает значение переменной на единицу (4) уменьшает значение переменной на два (5) в языке Си++ не существует	1	4
5.		При выполнении фрагмента кода <pre>int x = 3, y = 2, z = 1; if(x >= y) if(y <= z) std::cout << "Вариант 1"; else if(x >= z) std::cout << "Вариант 2"; else std::cout << "Вариант 3"; будет напечатано:</pre> (1) Вариант 1 (2) Вариант 2 (3) Вариант 3 (4) ничего не будет напечатано (5) программа не откомпилируется	2	8
6.	Задание открытого типа	Как подключить стандартную библиотеку <code>iostream</code> ?	<code>#include <iostream></code>	4
7.		Отметьте константу в следующем фрагменте кода: <pre>int k = 3; const int a = 2; int m = k + a - 3;</pre>	<code>a</code>	4
8.		Чему будет равен результат вычисления выражения: <pre>int d=5; bool b = true, c; c = (!b (d>3));</pre>	<code>true</code>	4
9.		Если <code>int n=3</code> , какой будет результат ? <pre>switch(n) { case 2: cout << "aaa"; break; case 3: cout << "ббб"; break; default: cout << "ввв"; break; }</pre>	<code>ббб</code>	4
10.	Задание комбинированного типа	Чему будет равен результат вычисления и почему: <pre>int i, k = 2, m = 10; i = (m)/(m/k - 5);</pre>	Ошибка выполнения, т.к. на ноль делить нельзя	4
ПК-3. Способен обеспечивать заданный уровень производительности, надежности и безопасности при создании вариантов архитектуры программного средства				
11.	Задание закрытого типа	В каком случае выражение вычислится быстрее: (1) <code>X = X + Y</code> (2) <code>X += Y</code>	2	4

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		(3) $X = Y + X$		
12.		Для чего предназначен фрагмент текста из заголовочного файла: <pre>//Пример файла test.h #ifndef TEST #define TEST //прочие строки кода #endif</pre> (1) для целей отладки (2) для определения символьной константы Test (3) для защиты от повторного включения файла test.h (4) для защиты от копирования (5) для защиты от удаления	3	10
13.		Какое определение функции является правильным? (1) <code>int f(int b,int c) {return;}</code> (2) <code>void f(int b,int c) {return b+c;}</code> (3) <code>int f(int b,int c) {return 0;}</code>	3	4
14.		Что будет выведено в стандартный поток в результате выполнения программы <pre>int main() { int x1 = 5; x1 = x2 - 2; int x2 = x1; cout << x2; return 0; }</pre> (1) -2 (2) 5 (3) 0 (4) ошибка компиляции	4	4
15.	Задание комбинированного типа	В каких выражениях произойдет заикливание программы и почему? (1) <code>for (int iCount = 0; iCount <= 4;);</code> (2) <code>while (true);</code> (3) <code>while (false);</code>	1, т.к. после каждой итерации переменная iCount не меняет своего значения и выражение остается истинно, 2, т.к. цикл, который начинается с <code>while(true)</code> выполняется бесконечно, пока не будет выполнена команда <code>break</code> .	4
16.	Задание открытого типа	В чем заключается назначение оператора перехода <code>goto</code> ?	изменяет последовательность выполнения операторов в программе	4

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
17.		Что будет выведено в стандартный поток в результате выполнения программы <pre>int main() { int x1 = 5; x1 = x2 - 2; int x2 = x1; cout << x2; return 0; }</pre>	Ошибка компиляции	4
18.		Если в массиве A 132 элемента, каким будет правильное обращение к последнему элементу массива?	A[131]	4
19.		Какой будет результат у данного выражения? <pre>int fl(int & x1, int *x2) { return ++x1 + ++(*x2); } int main() { int a = 7, k = 1; k = fl(a, &k); cout << a << " " << k; }</pre>	8 10	4
20.		В какой строке(строках) ошибка? <pre>1 int fl() { 2 char ch, ch1='a', ch2='b', ch3='c'; 3 ch=ch1+ch2+ch3; 4 fl=int(ch); 5 return fl; 6 }</pre>	4 и 5	4
ПК-4. Способен реализовывать программные средства				
21.	Задание закрытого типа	Исходя из данного кода какое высказывание верно? <pre>int main() { int a,b,c,d; a=1; b=2; c=a+b+p; cout << p; ... }</pre> (1) код не верен, потому что переменным c и d не присвоены значения (2) код верен, потому что по умолчанию все переменные имеют целочисленный тип (3) код не верен, потому что переменная p не объявлена	3	8
22.		В каких выражениях используются бинарные арифметические операции? (1) X + Y (2) c % d + 2 (3) xx * Y (4) xx++	1,2,3	4
23.		Если i = 5, какой будет результат? while (i <=5) { cout << (--i)-- << " ";	7	5

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<pre>if (i < 2) break; }</pre> (1) ошибка компиляции (2) цикл ни разу не будет выполнен (3) цикл будет выполняться бесконечно (4) 4 3 2 1 (5) 4 3 2 (6) 4 2 1 (7) 4 2		
24.		Что означает запись while (false);? (1) бесконечный цикл (2) цикл, который не выполняется ни разу (3) ошибка компиляции (4) аварийный выход из программы	2	3
25.		Для чего предназначен оператор namespace? (1) для заключения в группу объявлений классов, переменных и функций в отдельный контекст со своим именем (2) для заключения в группу объявлений классов, переменных и функций для использования только в текущем модуле (3) для использования классов, переменных и функций из других модулей программы без использования заголовочных файлов	1	3
26.	Задание открытого типа	Что выведет следующая программа? <pre>#include <iostream> int main() { int i; for(i = 0; i < 9; i++) std::cout << i+1; return 0; }</pre>	123456789	5
27.		Что произойдет, если определение функции будет находиться в файле в двух местах с одинаковыми сигнатурами?	Ошибка компиляции	3
28.		Какова последовательность создания исполняемого файла: (1) Создать файл с исходным текстом программы, который будет иметь расширение .cpp. (2) Скомпоновать объектный файл с необходимыми библиотеками. (3) Скомпилировать исходный код.	1-3-2	3
29.		Возможно ли использование подобного выражения? <pre>#define myincl "D:\Project\Coding\CodingU.h" #include myincl</pre>	Да	3
30.	Задание комбинированного типа	Чему равно значение выражения !((1 0) && 0) ? Как получили ответ?	1 1 0 = 1 (1 0) && 0 = 0 !((1 0) && 0) = 1	3
ПК-5. Способен участвовать в разработке операционной системы				
31.	Задание закрытого типа	Какое из следующих утверждений верно для этого кода? <pre>#include<stdio.h> int main() {</pre>	2,3	6

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<pre>int x = 10, y = 100%90, i; for(i=1; i<10; i++) if(x != y); printf("x = %d y = %d\n", x, y); return 0; }</pre> 1 : Функция printf() будет выполнена 10 раз. 2 : Выход программы будет x = 10 y = 10 3 : ; след if(x!=y) не приведет к ошибке. 4 : Программа не будет иметь никакого выхода.		
32.		Что описывает данный программный код? <pre>struct { char fio[30]; int date, code; double salary; }staff[100], *ps;</pre> Ответ: (1) определение массива структур и указателя на структуру (2) копирование строки salary в строку staff (3) создание указателя на строку date (4) создание итератора с указателем на строку date	1	5
33.		Для получения адреса переменной используется операция (1) * (2) & (3) -> (4) . (5) нет правильного ответа	2	3
34.		Какими по умолчанию объявляются элементы структуры?	2	3
35.		(1) private (2) public (3) protected (4) по умолчанию не объявляются		
36.	Задание открытого типа	Какой результат следующего выражения ? int* a; int b; a = &b; b = 7; *a++; cout << b;	7	3
37.		Какая операция позволяет получить значение, записанное по адресу, который содержится в указателе?	*	3
38.		Какой будет результат ? <pre>int f(int& x) { static int a = 0; if (!a) a = ++x; return a;} int y = 6; f(y); cout << f(y) << y;</pre>	77	5
39.		Что будет выведено на экран, если вызвать данную функцию последовательно три раза? <pre>void f1() { static int flag=0; if (!flag) { cout << "false "; flag=5; } else { cout << "true ";</pre>	false true false	7

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<pre>flag=0; } }</pre>		
40.		Что нужно сделать для освобождения памяти после выполнения такого кода ? <pre>char *a; a = new char[20];</pre>	delete [] a;	3
41.	Задание комбинированного типа	Какой массив занимает наибольший объем памяти? Обоснуйте ответ. <pre>char *s1= "QWERTY"; char s2[7]= "qwerty"; char *s3= "127*27";</pre> (1) Все массивы имеют одинаковый размер (2) s1 (3) s2 (4) s3	1, т.к. количество символов в трех строковых переменных одинаково	3

Полный комплект оценочных материалов по дисциплине (модулю) (фонд оценочных средств) хранится в электронном виде на кафедре, утверждающей рабочую программу дисциплины (модуля), и в Центре мониторинга и аудита качества обучения.

7.4. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине (модулю)

Таблица 10. Технологическая карта рейтинговых баллов по дисциплине (модулю)

№ п/п	Контролируемые мероприятия	Количество мероприятий / баллы	Максимальное количество баллов	Срок представления
2 семестр				
Основной блок				
1.	<i>Выполнение лабораторных работ</i>	2/10	20	Сроки указаны в Moodle
2.	<i>Контрольная работа</i>	1/20	20	Сроки указаны в Moodle
3.	<i>Коллоквиум</i>	2/25	50	
Всего			90	-
Блок бонусов				
4.	<i>Посещение занятий</i>		10	
Всего			10	-
ИТОГО			100	-
3 семестр				
Основной блок				
	<i>Выполнение лабораторных работ</i>	2/10	20	Сроки указаны в Moodle
	<i>Контрольная работа</i>	1/20	20	Сроки указаны в Moodle
	<i>Коллоквиум</i>	2/25	50	
Всего			90	-
Блок бонусов				
	<i>Посещение занятий</i>		10	
Всего			10	-
ИТОГО			100	-

Таблица 11. Шкала перевода рейтинговых баллов в итоговую оценку за семестр по дисциплине

Сумма баллов	Оценка по 4-балльной шкале
90–100	5 (отлично)
85–89	4 (хорошо)
75–84	
70–74	

Сумма баллов	Оценка по 4-балльной шкале
65–69	3 (удовлетворительно)
60–64	
Ниже 60	2 (неудовлетворительно)

При реализации дисциплины в зависимости от уровня подготовленности обучающихся могут быть использованы иные формы, методы контроля и оценочные средства, исходя из конкретной ситуации.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

8.1. Основная литература

1. Hopcroft J. E., Motwani R., Ullman J. D. Introduction to Automata Theory, Languages, and Computation (3rd Edition). — Addison-Wesley, Boston, MA, USA, 2006. — 750 с.
2. Шень А. Программирование: теоремы и задачи. — М.: МЦНМО, 2014. — 296 с.
3. Шень А., Верещагин Н. Языки и исчисления. — М.: МЦНМО, 2012. — 240 с.
4. Верещагин, Н. К. Колмогоровская сложность и алгоритмическая случайность [Электронный ресурс] / Н. К. Верещагин, В. А. Успенский, А. Шень. — Электрон. дан. — СПб: Лань, 2013. — 575 с. — Режим доступа: <https://e.lanbook.com/book/56395> — Загл. с экрана.

8.2. Учебно-методическое обеспечение для самостоятельной работы обучающихся:

1. Кривцова, И. Е. Основы дискретной математики. Часть 1. Учебное пособие [Электронный ресурс] / И. Е. Кривцова, И. С. Лебедев, А. В. Настека. — Электрон. дан. — СПб: ИТМО, 2016. — 92 с. — Режим доступа: http://books.ifmo.ru/book/1869/osnovy_diskretnoy_matematiki_chast_1_uchebnoe_posobie.htm — Загл. с экрана.

8.3. Дополнительная литература

1. Вики-конспекты. — http://neerc.ifmo.ru/wiki/index.php?title=Заглавная_страница

8.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимый для освоения дисциплины

1. Электронный каталог Научной библиотеки АГУ на базе MARK SQL НПО «Информ-систем»: <https://library.asu.edu.ru>
2. Корпоративный проект Ассоциации региональных библиотечных консорциумов (АРБИКОН) «Межрегиональная аналитическая роспись статей» (МАРС): <http://mars.arbicon.ru>
3. Единое окно доступа к образовательным ресурсам <http://window.edu.ru>

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для проведения лекционных занятий:

1. Используется аудитория, оборудованная необходимым количеством столов, стульев, доской маркерной и электронной.
2. Аудитория должна иметь следующие нормы освещенности
 - СНиП 23-05-95 «Естественное и искусственное освещение» норма освещенности аудиторий ВУЗов 400 Лк.
 - СанПиН 2.2.1/2.1.1.1278-03 «Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий» пункт 3.3.3. «Общее освещение в помещениях общественных зданий должно быть равномерным».
3. Электронная доска должна быть подключена к сети Интернет.

Для проведения лабораторных занятий:

1. Лабораторные занятия проводятся с группами или подгруппами не более 15 человек.

2. Аудитория должна быть оснащена необходимым количеством столов, стульев, доской маркерной и электронной.
4. Аудитория должна иметь следующие нормы освещенности
 - СНиП 23-05-95 «Естественное и искусственное освещение» норма освещенности аудиторий ВУЗов 400 Лк.
 - СанПиН 2.2.1/2.1.1.1278-03 «Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий» пункт 3.3.3. «Общее освещение в помещениях общественных зданий должно быть равномерным».
5. В аудитории должно быть не менее 15 компьютеров, находящихся в исправном состоянии.
6. Расположение компьютеров в аудитории должно позволять преподавателю подойти к рабочему месту студента.
7. Компьютеры должны быть соединены локальной сетью со скоростью не менее 1 Гбит/с и подключены к сети Интернет.
8. Компьютеры должны обладать минимальными характеристиками:
 - Объем оперативной памяти 16 Гб
 - Накопитель SDD 500 Гб
 - Процессор 12th Gen Intel(R) Core(TM) i3-12100
 - Видеоадаптер Intel(R) UHD Graphics 730

10. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ (МОДУЛЯ) ПРИ ОБУЧЕНИИ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Рабочая программа дисциплины (модуля) при необходимости может быть адаптирована для обучения (в том числе с применением дистанционных образовательных технологий) лиц с ограниченными возможностями здоровья, инвалидов. Для этого требуется заявление обучающихся, являющихся лицами с ограниченными возможностями здоровья, инвалидами, или их законных представителей и рекомендации психолого-медико-педагогической комиссии. При обучении лиц с ограниченными возможностями здоровья учитываются их индивидуальные психофизические особенности. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с нарушением слуха возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий, напечатанные увеличенным шрифтом), на аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и тифлосурдопереводчиков. Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания. Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала (понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т.д.). Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с нарушением зрения допускается аудиальное предоставление информации, а также использование на аудиторных занятиях звукозаписывающих устройств (диктофонов и т.д.). Допускается присутствие на занятиях ассистента (помощника), оказывающего обучающимся необходимую техническую помощь. Текущий контроль успеваемости осуществляется в устной форме. При проведении промежуточной аттестации для лиц с нарушением зрения тестирование может быть заменено на устное собеседование по вопросам.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, на аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается

присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитать задание, оформить ответ, общаться с преподавателем).