

Частное образовательное учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ
УПРАВЛЕНИЯ И ЭКОНОМИКИ»

РАССМОТРЕНО И ОДОБРЕНО

УТВЕРЖДАЮ

На заседании кафедры
информационных технологий и
математики
Протокол № 9 от 25.05.2023 г.

Первый проректор
С.В. Авдашкевич
28.06.2023

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дисциплина:	Б1.О.22 Алгоритмизация и программирование
Направление подготовки:	09.03.03 Прикладная информатика
Направленность (профиль):	Прикладная информатика в экономике
Уровень высшего образования:	Бакалавриат
Форма обучения:	очная, заочная
Разработчики:	Кандидат технических наук, доцент Курлов В.В. Старший преподаватель Смирнова С. Л.

Санкт-Петербург
2023

1. Цели и задачи дисциплины:

Цель освоения дисциплины:

формирование у студентов достаточно полного и строгого представления об алгоритмах программирования и о современных языках программирования (на примере языка C++).

Задачи дисциплины:

- способствовать освоению студентами теоретических основ алгоритмизации задач, практических приемов структурирования данных и программирования на алгоритмических языках высокого уровня (на примере языка C++);
- развить у студентов умения и навыки проектирования и разработки программных систем с использованием структурной и объектно-ориентированной технологий программирования.

2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы высшего образования

Планируемые результаты освоения ОП ВО (код и содержание компетенций)	Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Примечание
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;	ОПК-2.1 Знать современные информационные технологии (ИТ) и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.	-
	ОПК-2.2 Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.	
	ОПК-2.3 Владеть навыками использования современных информационных технологий (ИТ) и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.	
ОПК-3 Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;	ОПК-3.1 Знать принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	-
	ОПК-3.2 Уметь решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	
	ОПК-3.3 Владеть навыками подготовки обзоров, аннотаций, составления рефератов, докладов, публикаций и библиографии с учетом требований информационной безопасности	
ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью;	ОПК-4.1 Знать основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы (ИС).	-
	ОПК-4.2 Уметь применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.	
	ОПК-4.3 Владеть навыками составления технической документации на различных этапах жизненного цикла информационной системы.	

Планируемые результаты освоения ОП ВО (код и содержание компетенций)	Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Примечание
ОПК-5 Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;	ОПК-5.1 Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем.	-
	ОПК-5.2 Уметь выполнять параметрическую настройку информационных и автоматизированных систем.	
	ОПК-5.3 Владеть навыками инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем.	
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения;	ОПК-7.1 Знать основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и техно-логий.	-
	ОПК-7.2 Уметь применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.	
	ОПК-7.3 Владеть навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.	

Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Планируемые результаты обучения по дисциплине
ОПК-2.1. Знать современные информационные технологии (ИТ) и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.	Знать назначение стандартных инструментальных средств разработки программного обеспечения.
ОПК-2.2. Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.	Уметь выбирать современные информационные технологии и программные средства при решении задач профессиональной деятельности. Уметь находить и анализировать техническую документацию по использованию инструментальных средств разработки программного обеспечения.
ОПК-2.3. Владеть навыками использования современных информационных технологий (ИТ) и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.	Владеть навыками использования современных инструментальных средств при программировании на языке C++.
ОПК-3.1. Знать принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	Знать общие принципы построения алгоритмов, основные алгоритмические конструкции. Знать базовые понятия и элементы языка программирования C++.
ОПК-3.2. Уметь решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	Уметь реализовывать алгоритмы решения задач на языке программирования C++.

Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Планируемые результаты обучения по дисциплине
ОПК-3.3. Владеть навыками подготовки обзоров, аннотаций, составления рефератов, докладов, публикаций и библиографии с учетом требований информационной безопасности	Владеть навыками чтения и понимания программ, реализующих несложные алгоритмы обработки числовых и текстовых данных. Владеть навыками подготовки обзоров, докладов с учетом знаний в области алгоритмизации и программирования.
ОПК-4.1. Знать основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы (ИС).	Знать основные правила форматирования кода на языке программирования C++.
ОПК-4.2. Уметь применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.	Уметь использовать комментарии в программном коде как способ первичного документирования программы.
ОПК-4.3. Владеть навыками составления технической документации на различных этапах жизненного цикла информационной системы.	Владеть навыками форматирования кода и его комментирования на языке программирования C++.
ОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем.	Знать современные программные средства для практической реализации программного обеспечения на персональных компьютерах.
ОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем.	Уметь осуществлять установку, настройку инструментальных средств разработки программного обеспечения для решения задач на языке C++.
ОПК-5.3. Владеть навыками установки программного и аппаратного обеспечения информационных и автоматизированных систем.	Владеть навыками установки инструментальных средств разработки программного обеспечения для решения задач на языке C++.
ОПК-7.1. Знать основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.	Знать основы структурного программирования на языке C++. Знать основы объектно-ориентированного программирования на языке C++.
ОПК-7.2. Уметь применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.	Уметь создавать программы на языке C++ для решения типовых задач базового уровня из различных предметных областей.
ОПК-7.3. Владеть навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.	Владеть практическими навыками составления алгоритмов для решения задач и их реализации на языке программирования C++. навыками тестирования и отладки программ.

3. Содержание, объем дисциплины и формы проведения занятий

№ п/п	Наименование темы дисциплины	Компетенции	Оценочные средства текущего контроля		
			ЗНАТЬ	УМЕТЬ	ВЛАДЕТЬ
			ОПК-2.1 ОПК-3.1 ОПК-4.1 ОПК-5.1 ОПК-7.1	ОПК-2.2 ОПК-3.2 ОПК-4.2 ОПК-5.2 ОПК-7.2	ОПК-2.3 ОПК-3.3 ОПК-4.3 ОПК-5.3 ОПК-7.3
1	Основы алгоритмизации. Общие принципы разработки программного обеспечения.	ОПК-2 ОПК-3 ОПК-4 ОПК-5	Доклад, сообщение/ Реферат №1 (10) Задача №1 (10) Собеседование, опрос/ Контрольная работа №1 (10)	Задача №1 (10)	Расчетно-графическая работа №1 (20)

№ п/п	Наименование темы дисциплины	Компетенции	Оценочные средства текущего контроля		
			ЗНАТЬ	УМЕТЬ	ВЛАДЕТЬ
			ОПК-2.1 ОПК-3.1 ОПК-4.1 ОПК-5.1 ОПК-7.1	ОПК-2.2 ОПК-3.2 ОПК-4.2 ОПК-5.2 ОПК-7.2	ОПК-2.3 ОПК-3.3 ОПК-4.3 ОПК-5.3 ОПК-7.3
2	Введение в программирование на языке C++.	ОПК-3 ОПК-7	Задача №2 (10) Собеседование, опрос/ Контрольная работа №1 (10)	Задача №2 (10)	Расчетно-графическая работа №1 (20)
3	Управление вычислительным процессом.	ОПК-3 ОПК-7	Задача №3 (10) Задача №4 (10) Задача №5 (10) Задача №6 (10)	Задача №3 (10) Задача №4 (10) Задача №5 (10) Задача №6 (10)	Расчетно-графическая работа №1 (20)
Количество баллов (100 баллов):			100		
4	Производные типы данных.	ОПК-3 ОПК-7	Задача №10 (10) Задача №7 (10) Задача №8 (10) Задача №9 (10) Собеседование, опрос/ Контрольная работа №2 (10)	Задача №10 (10) Задача №7 (10) Задача №8 (10) Задача №9 (10)	Расчетно-графическая работа №2 (20)
5	Элементы модульного программирования.	ОПК-3 ОПК-7	Задача №11 (10) Собеседование, опрос/ Контрольная работа №3 (10)	Задача №11 (10)	Расчетно-графическая работа №2 (20)
6	Потоковый ввод-вывод.	ОПК-3 ОПК-7	Задача №12 (10)	Задача №12 (10)	Расчетно-графическая работа №2 (20)
Количество баллов (100 баллов):			100		
7	Элементы профессионального стиля программирования: стандарты и рекомендации.	ОПК-2 ОПК-4 ОПК-7	Расчетно-графическая работа №3 (20)	Расчетно-графическая работа №3 (20)	Расчетно-графическая работа №3 (20)
8	Абстрактные структуры данных.	ОПК-3 ОПК-7	Задача №13 (10) Задача №14 (10) Расчетно-графическая работа №3 (20) Собеседование, опрос/ Контрольная работа №4 (10)	Задача №13 (10) Задача №14 (10) Расчетно-графическая работа №3 (20)	Расчетно-графическая работа №3 (20)

№ п/п	Наименование темы дисциплины	Компетенции	Оценочные средства текущего контроля		
			ЗНАТЬ	УМЕТЬ	ВЛАДЕТЬ
			ОПК-2.1 ОПК-3.1 ОПК-4.1 ОПК-5.1 ОПК-7.1	ОПК-2.2 ОПК-3.2 ОПК-4.2 ОПК-5.2 ОПК-7.2	ОПК-2.3 ОПК-3.3 ОПК-4.3 ОПК-5.3 ОПК-7.3
9	Алгоритмы поиска и сортировки.	ОПК-3 ОПК-7	Расчетно-графическая работа №3 (20) Собеседование, опрос/ Контрольная работа №4 (10)	Расчетно-графическая работа №3 (20)	Расчетно-графическая работа №3 (20)
10	Элементы объектно-ориентированного программирования и их реализация на языке C++.	ОПК-3 ОПК-4 ОПК-7	Задача №15 (10) Задача №16 (10) Задача №17 (10) Задача №18 (10) Задача №19 (10) Расчетно-графическая работа №3 (20) Собеседование, опрос/ Контрольная работа №4 (10)	Задача №15 (10) Задача №16 (10) Задача №17 (10) Задача №18 (10) Задача №19 (10) Расчетно-графическая работа №3 (20)	Расчетно-графическая работа №3 (20)
Количество баллов (100 баллов):			100		

Содержание учебного материала, лабораторные работы и практические занятия, курсовая работа
<p>Тема 1: Основы алгоритмизации. Общие принципы разработки программного обеспечения. Понятие алгоритма и его основные свойства: массовость, дискретность, детерминированность, результативность. Основные этапы разработки алгоритмов: постановка задачи, построение математической модели, разработка алгоритма решения задачи, проверка правильности и оценка сложности алгоритма. Формы и способы представления алгоритмов. Правила построения алгоритмов из базовых алгоритмических конструкций. Типы алгоритмических процессов: линейные, ветвящиеся, циклические. Основные этапы решения задач на ЭВМ: постановка задачи и ее формализация, выбор структур данных и разработка алгоритма, реализация структур данных и алгоритма на языке программирования, отладка, тестирование и документирование программы. Понятие технологии разработки программного продукта и структура его жизненного цикла. Понятие и структура системы программирования. Последовательность обработки программы от исходного текста на языке высокого уровня до исполняемого кода. Назначение и функции транслятора. Компиляторы и интерпретаторы. Основные этапы трансляции программы. Назначение и функции компоновщика. Схема функционирования редактора связей. Структура объектного и загрузочного модуля. Назначение и функции загрузчика прикладных программ. Методы трансляции адресов программы. Библиотеки подпрограмм как составная часть системы программирования. Статические и динамические библиотеки. Понятие мобильности и обеспечение переносимости программных продуктов. История развития и классификация языков программирования. Краткий обзор современных методологий программирования. Сравнительная характеристика языков программирования высокого уровня. История создания языков программирования C и C++. Обзор языков программирования семейства C. Структура алгоритмического языка. Понятие синтаксиса, семантики, прагматики и лексики. Понятие лексемы и основные лексические структуры языка. Идентификаторы и служебные слова. Переменные и константы. Целочисленные и вещественные константы. Символьные константы и управляющие последовательности. Строковые константы. Структура и компоненты консольного приложения. Основные этапы подготовки исполняемого файла. Особенности препроцессорного преобразования программы. Стандартная библиотека. Заголовочные файлы и их структура.</p> <p>Практические занятия/самостоятельная работа:</p>

Содержание учебного материала, лабораторные работы и практические занятия, курсовая работа
<p>Анализ и составление алгоритмических конструкций. Использование инструментальных средств разработки на языке C++. Настройка среды разработки. Чтение диагностических сообщений.</p> <p>Лабораторная работа: -</p> <p>Тема 2: Введение в программирование на языке C++.</p> <p>Концепция типа данных в языках программирования. Классификация типов данных языка C++. Фундаментальные и производные типы. Переменные в языке C++: определение и инициализация. Идентификаторы и модификаторы базовых типов. Арифметические типы данных. Формы внутреннего представления данных с плавающей запятой. Представление логических данных. Определение и инициализация символьных переменных. Кодовые таблицы. Тип void. Понятие области видимости переменной. Локальные и глобальные переменные. Операция доступа к области видимости. Объявление и инициализация именованных констант. Выражения и операции в языке программирования C++. Классификация операций. Приоритет и порядок выполнения операций. Правила преобразования типов данных в выражениях. Операция приведения типа. Стандартные математические функции и их использование в выражениях. Функции преобразования данных. Обработка символов. Функции классификации и преобразования символов. Генерирование случайных чисел в заданном промежутке. Локализация и интернационализация. Возможности интернационализации. Установка локали.</p> <p>Практические занятия/самостоятельная работа: Написание программы на C++, отладка, тестирование, устранение дефектов.</p> <p>Лабораторная работа: -</p>
<p>Тема 3: Управление вычислительным процессом.</p> <p>Основные принципы структурного программирования. Структурная теорема Бёма-Якопини. Базовые управляющие структуры: следование, ветвление, цикл. Общая характеристика и классификация операторов языка. Пустой оператор. Операторы управления работой программы. Составные операторы. Операторы ветвления. Операторы циклов. Операторы перехода.</p> <p>Практические занятия/самостоятельная работа: Алгоритмы ветвления. Циклические алгоритмы. Алгоритмы обработки одномерных числовых последовательностей.</p> <p>Практические занятия/самостоятельная работа: Написание программы на C++, отладка, тестирование, устранение дефектов.</p> <p>Лабораторная работа: -</p>
<p>Тема 4: Производные типы данных.</p> <p>Общая характеристика типов данных, определяемых пользователем. Массивы. Индексирование элементов массива. Определение и инициализация элементов одномерного массива. Многомерный массив. Размещение элементов многомерного массива в памяти и организация доступа к ним. Инициализация элементов многомерного массива. Типовые задачи обработки массивов. Представление строки как одномерного массива символов. Особенности определения и инициализации строковых переменных. Типовые задачи обработки строк. Использование функций работы со строками стандартной библиотеки (cstring). Статические и динамические переменные. Адреса и указатели. Объявление и инициализация указателей. Определение адреса переменной в памяти. Получение значения переменной по указанному адресу. Понятие нулевого адреса. Адресная арифметика. Связь между указателями и массивами. Механизм управления динамическим распределением памяти. Распределение одномерных и многомерных массивов в памяти. Структуры, прямой и косвенный доступ к элементам. Массивы и структуры в качестве элементов структур. Указатели в структурах. Массивы структур. Понятие объединения и его отличие от структуры.</p> <p>Практические занятия/самостоятельная работа: Написание программы на C++, отладка, тестирование, устранение дефектов.</p> <p>Лабораторная работа: -</p>
<p>Тема 5: Элементы модульного программирования.</p>

Содержание учебного материала, лабораторные работы и практические занятия, курсовая работа

Основные принципы модульного программирования. Общие сведения о функциях в языке C++. Структура функции и область ее видимости. Тип значения, возвращаемого функцией и оператор return. Формальные и фактические параметры. Прототипы функций и их описание в заголовочных файлах. Передача параметров по значению. Передача параметров по адресу. Передача параметров по ссылке. Использование квалификатора const для защиты аргументов. Вызов функции и возвращаемое значение. Стек вызовов. Рекурсивный вызов функций. Область действия и область видимости идентификаторов. Классы памяти. Автоматические и регистровые переменные. Локальные и глобальные статические переменные. Использование статических переменных в функциях. Понятие внешней переменной. Понятие пространства имен и его объявление. Пространство имен стандартной библиотеки. Массивы и структуры как формальные параметры функций. Функции с переменным числом параметров. Организация доступа к списку параметров через указатель. Создание и использование указателей на функции. Передача функции в качестве параметра. Функции быстрой сортировки и бинарного поиска стандартной библиотеки (cstdlib). Аргументы функции main(). Перегрузка функций. Шаблоны функций. Препроцессорные средства: замены, включения, условная компиляция, стандартные макроопределения. Разработка крупных программ. Структуризация программ с помощью модулей. Раздельная компиляция.

Практические занятия/самостоятельная работа:

Написание программы на C++, отладка, тестирование, устранение дефектов.

Лабораторная работа: -**Тема 6: Поточковый ввод-вывод.**

Особенности организации ввода-вывода в языке C++. Понятие потока и его свойства. Стандартная библиотека ввода-вывода. Форматированный ввод-вывод в C++. Файловые потоки. Текстовые и двоичные потоки. Технология работы с файлами в программах.

Практические занятия/самостоятельная работа:

Написание программы на C++, отладка, тестирование, устранение дефектов.

Лабораторная работа: -**Тема 7: Элементы профессионального стиля программирования: стандарты и рекомендации.**

Подходы к разработке программ. Проектирование сверху-вниз и снизу-вверх. Структуризация программы на уровне блоков, функций, файлов исходного кода. Вопросы стиля программирования. Форматирование кода. Комментирование. Контроль возвращаемых значений функций. Приемы программирования. Взаимодействие с операционной средой: стек финальных функций, доступ к переменным среды. Функции работы со временем и датой. Временная отметка и структура времени tm. Функции преобразования и форматирования даты/времени.

Практические занятия/самостоятельная работа:**Практические занятия/самостоятельная работа:**

Написание программы на C++, отладка, тестирование, устранение дефектов.

Лабораторная работа: -**Тема 8: Абстрактные структуры данных.**

Понятие абстрактного типа данных и его взаимосвязь с понятием структуры. Основные структуры хранения данных и особенности их размещения в оперативной памяти. Проблема отображения абстрактных структур данных на структуры хранения. Классификация абстрактных структур данных. Линейные и нелинейные, статические и динамические структуры. Структуры с прямым и последовательным доступом. Основные линейные абстрактные структуры данных: список, очередь, стек. Виды линейных списков: односвязный, двусвязный, кольцевой. Структура элементов односвязного и двусвязного списков. Реализация основных операций над линейными списками. Формирование и организация доступа к элементам линейного списка. Включение, удаление и просмотр элементов односвязного и двусвязного списков. Использование рекурсии при работе с линейными динамическими списками. Векторное представление очереди и стека. Представление очереди и стека на основе линейного списка. Реализация основных операций с очередью и стеком. Нелинейные структуры данных. Общие сведения о деревьях. Представление деревьев в оперативной памяти с помощью связанных списков. Понятие бинарного дерева. Организация рекурсивного обхода вершин. Формирование бинарного дерева поиска. Сбалансированные бинарные деревья поиска. Оценка вычислительной сложности операций поиска в бинарных деревьях. Понятие об оптимальных деревьях поиска. Реализация основных операций над бинарными деревьями.

Практические занятия/самостоятельная работа:

Написание программы на C++, отладка, тестирование, устранение дефектов.

Лабораторная работа: -**Тема 9: Алгоритмы поиска и сортировки.**

Содержание учебного материала, лабораторные работы и практические занятия, курсовая работа
<p>Общая характеристика алгоритмов сортировки данных и основные требования, предъявляемые к ним. Внутренняя и внешняя сортировка. Основные классы методов сортировки: обменная сортировка, сортировка выбором, сортировка вставками, сортировка слиянием. Примеры методов сортировки. Общая характеристика алгоритмов поиска. Линейный поиск в неупорядоченных/упорядоченных последовательностях. Бинарный поиск в упорядоченной последовательности. Оценка вычислительной сложности алгоритмов поиска. Понятие о древовидных таблицах поиска и их структура. Таблицы с вычисляемыми входами. Выбор функции рандомизации. Алгоритмы вычисления хеш-функций. Понятие коллизии и методы ее разрешения.</p> <p>Практические занятия/самостоятельная работа: Написание программы на C++, отладка, тестирование, устранение дефектов.</p> <p>Лабораторная работа: -</p>
<p>Тема 10: Элементы объектно-ориентированного программирования и их реализация на языке C++. Основные понятия объектно-ориентированного программирования. Пространства имен. Механизм классов. Объекты, инкапсуляция. Управление доступом. Перегрузка функций и операторов. Наследование. Полиморфизм. Параметризованные типы и функции. Обработка исключений. Стандартная библиотека языка C++.</p> <p>Практические занятия/самостоятельная работа: Написание программы на C++, отладка, тестирование, устранение дефектов.</p> <p>Лабораторная работа: -</p>
<p>Курсовая работа: не предусмотрено учебным планом</p>

Очная форма обучения

Вид учебной работы	Всего часов	Семестр 2	Семестр 3	Семестр 4
Аудиторные занятия (АЗ):	138	48	54	36
Лекционные занятия (Лек)	52	16	18	18
Лабораторные занятия (Лаб)	0	0	0	0
Практические занятия (Пр)	86	32	36	18
Самостоятельная работа студента (СР)	170	55	82	33
Курсовая работа	0	0	0	0
Другие виды самостоятельной работы*	170	55	82	33
Контроль самостоятельной работы (КСР)	16	5	8	3
Контактная работа (КоР)	154	53	62	39
Форма промежуточной аттестации	0	Зачет	Зачет	Экзамен
Подготовка к экзамену и сдача экзамена (СР, КоР)	36	0	0	36
Общая трудоемкость дисциплины, часы/ЗЕТ	360/10	108/3	144/4	108/3

* Подготовка к аудиторным занятиям, подготовка к зачету (при наличии)

№	Наименование темы дисциплины	Семестр/Курс	Количество учебных часов				Практическая подготовка
			В том числе по видам аудиторных занятий			СР	
			Лек	Пр	Лаб		
1	Основы алгоритмизации. Общие принципы разработки программного обеспечения.	2	4	4	0	18	4
2	Введение в программирование на языке C++.	2	6	8	0	18	8
3	Управление вычислительным процессом.	2	6	20	0	19	20
Итого за 2 семестр:			16	32	0	55	32
4	Производные типы данных.	3	6	4	0	27	4
5	Элементы модульного программирования.	3	6	8	0	27	8
6	Потоковый ввод-вывод.	3	6	24	0	28	24
Итого за 3 семестр:			18	36	0	82	36
7	Элементы профессионального стиля программирования: стандарты и рекомендации.	4	4	4	0	9	4
8	Абстрактные структуры данных.	4	4	4	0	8	4
9	Алгоритмы поиска и сортировки.	4	4	4	0	8	4
10	Элементы объектно-ориентированного программирования и их реализация на языке C++.	4	6	6	0	8	6
Итого за 4 семестр:			18	18	0	33	18

09.03.03 Прикладная информатика, направленность (профиль) "Прикладная информатика в экономике"
 Рабочая программа дисциплины
 Дисциплина: Б1.О.22 Алгоритмизация и программирование
 Форма обучения: очная, заочная
 Разработана для приема 2021/2022, 2022/2023 учебного года
 Обновлено на 2023/2024 учебный год

№	Наименование темы дисциплины	Семестр/ Курс	Количество учебных часов				Практическая подготовка
			В том числе по видам аудиторных занятий			СР	
			Лек	Пр	Лаб		
Итого:			52	86	0	170	86

* Практическая подготовка при реализации дисциплин организована путем проведения практических занятий и (или) выполнения лабораторных и (или) курсовых работ и предусматривает выполнение работ, связанных с будущей профессиональной деятельностью.

Заочная форма обучения

Вид учебной работы	Всего часов	Семестр 3	Семестр 4	Семестр 5
Аудиторные занятия (АЗ):	22	4	8	10
Лекционные занятия (Лек)	10	2	4	4
Лабораторные занятия (Лаб)	0	0	0	0
Практические занятия (Пр)	12	2	4	6
Самостоятельная работа студента (СР)	306	95	126	85
Курсовая работа	0	0	0	0
Другие виды самостоятельной работы*	306	95	126	85
Контроль самостоятельной работы (КСР)	15	5	6	4
Контактная работа (КоР)	37	9	14	14
Форма промежуточной аттестации	0	Зачет	Зачет	Экзамен
Подготовка к экзамену/зачету и сдача экзамена/зачета (СР, КоР)	17	4	4	9
Общая трудоемкость дисциплины, часы/ ЗЕТ	360/10	108/3	144/4	108/3

* Подготовка к аудиторным занятиям

№	Наименование темы дисциплины	Семестр/ Курс	Количество учебных часов				Практическая подготовка
			В том числе по видам аудиторных занятий			СР	
			Лек	Пр	Лаб		
1	Основы алгоритмизации. Общие принципы разработки программного обеспечения.	3	2	0	0	31	4
2	Введение в программирование на языке C++.	3	0	0	0	31	8
3	Управление вычислительным процессом.	3	0	2	0	33	20
Итого за 3 семестр:			2	2	0	95	32
4	Производные типы данных.	4	2	0	0	42	4
5	Элементы модульного программирования.	4	0	2	0	42	8
6	Потоковый ввод-вывод.	4	2	2	0	42	24
Итого за 4 семестр:			4	4	0	126	36
7	Элементы профессионального стиля программирования: стандарты и рекомендации.	5	2	0	0	25	4
8	Абстрактные структуры данных.	5	0	2	0	20	4
9	Алгоритмы поиска и сортировки.	5	2	2	0	20	4
10	Элементы объектно-ориентированного программирования и их реализация на языке C++.	5	0	2	0	20	6
Итого за 5 семестр:			4	6	0	85	18
Итого:			10	12	0	306	86

* Практическая подготовка при реализации дисциплин организована путем проведения практических занятий и (или) выполнения лабораторных и (или) курсовых работ и (или) путем выделения часов из часов, отведенных на самостоятельную работу, и предусматривает выполнение работ, связанных с будущей профессиональной деятельностью.

4. Способ реализации дисциплины Без использования онлайн-курса.

5. Учебно-методическое обеспечение дисциплины:

Основная литература:

1. ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C++: ПРАКТИЧЕСКИЙ КУРС. Учебное пособие для вузов / Огнева М. В., Кудрина Е. В. - Национальный исследовательский Саратовский государственный университет имени Н.Г. Чернышевского (г. Саратов)., 2022 г. - 335 с. - ISBN 978-5-534-05123-0 – Режим доступа: <https://urait.ru/book/programmirovanie-na-yazyke-s-prakticheskiy-kurs-492984>

2. МАТЕМАТИЧЕСКАЯ ЛОГИКА И ТЕОРИЯ АЛГОРИТМОВ 5-е изд. Учебник и практикум для вузов / Судоплатов С. В., Овчинникова Е. В. - Новосибирский государственный технический университет (г. Новосибирск)., 2022 г. - 207 с. - ISBN 978-5-534-12274-9 – Режим доступа: <https://urait.ru/book/matematiceskaya-logika-i-teoriya-algoritmov-447321>

3. АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ. Учебник для вузов / Трофимов В. В., Павловская Т. А. ; Под ред. Трофимова В.В. - Санкт-Петербургский государственный экономический университет (г. Санкт-Петербург).; Национальный исследовательский университет ИТМО (г. Санкт-Петербург)., 2022 г. - 137 с. - ISBN 978-5-534-07834-3 – Режим доступа: <https://urait.ru/book/algoritmizaciya-i-programmirovanie-491215>

Дополнительная литература:

1. ТЕОРИЯ АЛГОРИТМОВ. ВВЕДЕНИЕ В СЛОЖНОСТЬ ВЫЧИСЛЕНИЙ 2-е изд., испр. и доп. Учебное пособие для вузов / Крупский В. Н. - Московский государственный университет имени М.В. Ломоносова (г. Москва)., 2022 г. - 117 с. - ISBN 978-5-534-04817-9 – Режим доступа: <https://urait.ru/book/teoriya-algoritmov-vvedenie-v-slozhnost-vychisleniy-492937>

2. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C#. Учебное пособие для вузов / Кудрина Е. В., Огнева М. В. - Национальный исследовательский Саратовский государственный университет имени Н.Г. Чернышевского (г. Саратов)., 2022 г. - 322 с. - ISBN 978-5-534-09796-2 – Режим доступа: <https://urait.ru/book/osnovy-algoritmizacii-i-programmirovaniya-na-yazyke-c-494874>

3. ПРОГРАММИРОВАНИЕ. ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД. Учебник и практикум для вузов / Зыков С. В. - Национальный исследовательский университет «Высшая школа экономики» (г. Москва)., 2022 г. - 155 с. - ISBN 978-5-534-00850-0 – Режим доступа: <https://urait.ru/book/programmirovanie-obektno-orientirovannyy-podhod-490423>

6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения

1. Операционная система
2. Пакет прикладных офисных программ
3. Антивирусное программное обеспечение
4. LMS Moodle
5. Вебинарная платформа
6. Code::Blocks (MinGW (Minimalist GNU for Windows))

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», информационных справочных систем и профессиональных баз данных, необходимых для освоения дисциплины

1. ibooks.ru : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://ibooks.ru>. - Текст: электронный

2. Электронно-библиотечная система СПБУТУиЭ : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <http://libume.ru>. - Текст: электронный

3. Юрайт : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://urait.ru>. - Текст: электронный

4. eLibrary.ru : научная электронная библиотека [Электронный ресурс] : профессиональная база данных. - Режим доступа: <http://elibrary.ru>. - Текст: электронный

5. Архив научных журналов НЭИКОН [Электронный ресурс] : профессиональная база данных. - Режим доступа: arhiv.neicon.ru. - Текст: электронный

6. КиберЛенинка : научная электронная библиотека [Электронный ресурс] : информационная справочная система. - Режим доступа: <http://cyberleninka.ru>. - Текст: электронный

7. Лань : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://e.lanbook.com>. - Текст: электронный

8. it-world.ru [Электронный ресурс] : информационная справочная система . - Режим доступа: <https://www.it-world.ru>. - Текст: электронный

9. Connect: IT-технологии : информационная справочная система. - Режим доступа: <https://www.connect-wit.ru/>. - Текст: электронный

10. Виртуальный компьютерный музей [Электронный ресурс] : информационная справочная система . - Режим доступа: <https://www.computer-museum.ru>. - Текст: электронный

11. Компьютерра : информационная справочная система . - Режим доступа: <https://www.computerra.ru/>. - Текст: электронный

12. Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации: профессиональная база данных. - Режим доступа: <https://digital.gov.ru>. - Текст: электронный

13. Федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций: профессиональная база данных . - Режим доступа: <https://rkn.gov.ru>. - Текст: электронный

14. Math-Net.Ru: профессиональная база данных . - Режим доступа: <https://www.mathnet.ru/>. - Текст: электронный

8. Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения занятий лекционного типа, семинарского типа - практических занятий, для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оборудованная: рабочими местами для обучающихся, оснащенными специальной мебелью; рабочим местом преподавателя, оснащенным специальной мебелью, персональным компьютером с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; техническими средствами обучения - мультимедийным оборудованием (проектор, экран, колонки) и маркерной доской; лицензионным программным обеспечением

2. Учебная аудитория для проведения занятий семинарского типа - практических занятий – компьютерный класс, оборудованный рабочими местами для обучающихся, оснащенными специальной мебелью, персональными компьютерами с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; рабочим местом преподавателя, оснащенным специальной мебелью, персональным компьютером с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; техническими средствами обучения - мультимедийным оборудованием (проектор, экран, колонки) и маркерной доской; лицензионным программным обеспечением

3. При применении электронного обучения, дистанционных образовательных технологий используются: виртуальные аналоги учебных аудиторий - вебинарные комнаты на вебинарных платформах, рабочее место преподавателя, оснащенное персональным компьютером (планшет, мобильное устройство) с возможностью подключения к сети «Интернет», доступом к электронной информационно-образовательной среде Университета и к информационно-образовательному порталу Университета imeos.ru, веб-камерой, микрофоном и гарнитурой (в т.ч. интегрированными в устройствами), программным обеспечением; рабочее место

обучающегося оснащено персональным компьютером (планшет, мобильное устройство) с возможностью подключения к сети «Интернет», доступом к электронной информационно-образовательной среде Университета и к информационно-образовательному portalу Университета imeos.ru, веб-камерой, микрофоном и гарнитурой (в т.ч. интегрированными в устройства). Авторизация на информационно-образовательном portalе Университета imeos.ru и начало работы осуществляются с использованием персональной учетной записи (логина и пароля). Лицензионное программное обеспечение

4. Помещение для самостоятельной работы, оборудованное специальной мебелью, персональными компьютерами с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета, лицензионным программным обеспечением

9. Оценочные материалы по дисциплине

Описание оценочных средств (показатели и критерии оценивания, шкалы оценивания) представлено в приложении к основной профессиональной образовательной программе «Каталог оценочных средств текущего контроля и промежуточной аттестации».

Процедуры оценивания знаний, умений, навыков и (или) опыта деятельности приводятся в соответствующих методических материалах и локальных нормативных актах Университета.

Для оценивания учебных достижений студентов в Университете действует балльно-рейтинговая система.

Если оценка, соответствующая набранной в семестре сумме рейтинговых баллов, удовлетворяет студента, то она является итоговой оценкой по дисциплине при проведении промежуточной аттестации в форме экзамена/зачета с оценкой/зачета.

Условием сдачи экзамена/зачета с оценкой/зачета с целью повышения итоговой оценки по дисциплине является сдача студентом экзамена, за который он получает экзаменационные баллы без учета баллов, полученных за текущий контроль:

Шкала оценивания учебных достижений по дисциплине, завершающейся зачетом без оценки

Баллы по дисциплине	60 и менее		61-73		74-90		91-100	
Итоговая оценка по дисциплине	Незачет		Зачет					
Баллы в международной шкале ECTS с буквенным обозначением уровня	50 и менее	51-60	61-67	68-73	74-83	84-90	91-100	
	F	Fx	E	D	C	B	A	
Уровень сформированности компетенций	Не сформированы		Пороговый		Высокий		Повышенный	

Шкала оценивания учебных достижений по дисциплине, завершающейся экзаменом/зачетом с оценкой

Баллы по дисциплине	60 и менее		61-73		74-90		91-100	
Итоговая оценка по дисциплине	Неудовлетворительно		Удовлетворительно		Хорошо		Отлично	
Баллы в международной шкале ECTS с буквенным обозначением уровня	<50	51-60	61-67	68-73	74-83	84-90	91-100	
	F	Fx	E	D	C	B	A	
Уровень сформированности компетенций	Не сформированы		Пороговый		Высокий		Повышенный	

9.1. Типовые контрольные задания для текущего контроля

Задача №1

1. Напишите название свойства, соответствующее каждому определению.

_____ - описание процесса решения на языке исполнителя.

_____ - алгоритм приводит к решению задачи.
_____ - алгоритм дает правильное решение при различных наборах начальных данных.

_____ - процесс решения разбит на отдельные шаги.
_____ - запись алгоритма четкая, полная, ясная.
_____ - задача решается за известное количество шагов.

2. Запишите исполнителей для приведённых ниже видов работ:

Уборка мусора во дворе –

Перевозка пассажиров в поезде –

Выдача заработной платы –

Приём экзаменов в школе –

Сдача экзамена в университете –

Набор текста на компьютере –

Приготовление еды в ресторане –

3. Определите полный набор данных для решения следующих задач обработки информации:

Вычисление стоимости покупок в магазине –

Вычисление суммы сдачи от данных вами продавцу денег –

Определение времени показа по телевизору интересующего вас фильма –

Вычисление площади треугольника –

Определение времени падения кирпича с крыши дома –

Определение месячной платы за расход электроэнергии –

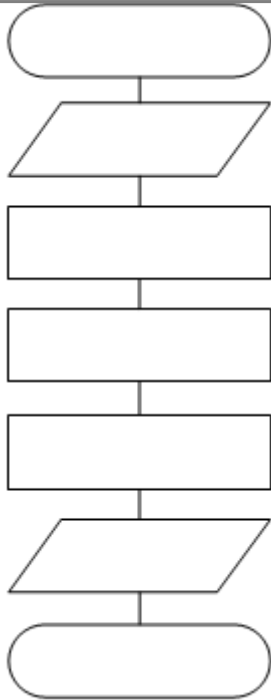
Перевод русского текста на транслит –

4. Есть исполнитель «Перевозчик», который перевозит через реку волка, козу и капусту. Напишите алгоритм перевоза через реку волка, козы и капусты, если СКИ «Перевозчика» содержит 5 команд: ВЗЯТЬ КОЗУ, ВЗЯТЬ ВОЛКА, ВЗЯТЬ КАПУСТУ, ВЫСАДИТЬ, ПЕРЕПЛЫТЬ. В лодку может поместиться только один предмет или животное. Нельзя оставлять на берегу одних волка с козой и козу с капустой.

5. Составить алгоритм в *словесной форме* для решения задачи нахождения минимального из трех значений.

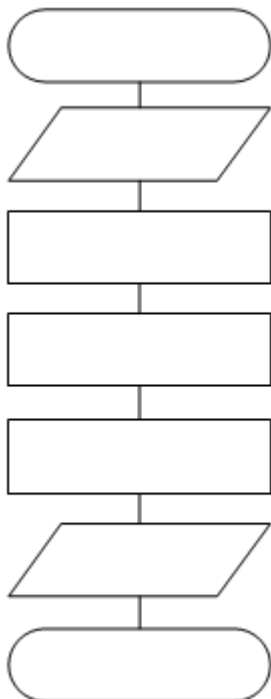
6. Составить алгоритм в *словесной форме* для решения задачи деления отрезка пополам с помощью циркуля и линейки.

7. Дана длина ребра куба. Составьте блок-схему алгоритма нахождения площади грани, полной поверхности и объема этого куба.



8. Дано a . Не используя никаких функций и операций, кроме умножения, получить a^8 за три операции. Заполните шаблон, используя таблицу значений при $a=2$.

шаг алгоритма	аргумент	промежуточные величины		результат	пооянения
	a	b	c	y	
1	2				ввод a
2		4			
3			16		
4				256	
5					вывод $y=256$
6					конец



9. Дано действительное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить:

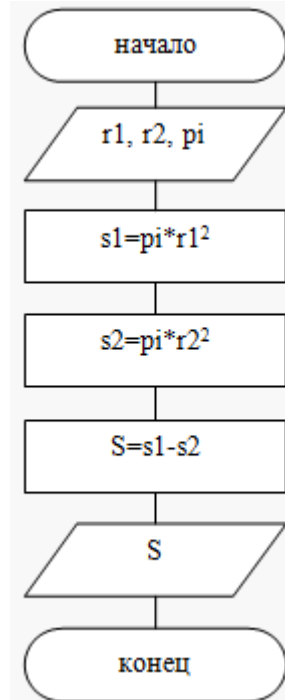
а) a^4 за две операции;

б) a^6 за три операции;

в) a^7 за четыре операции;

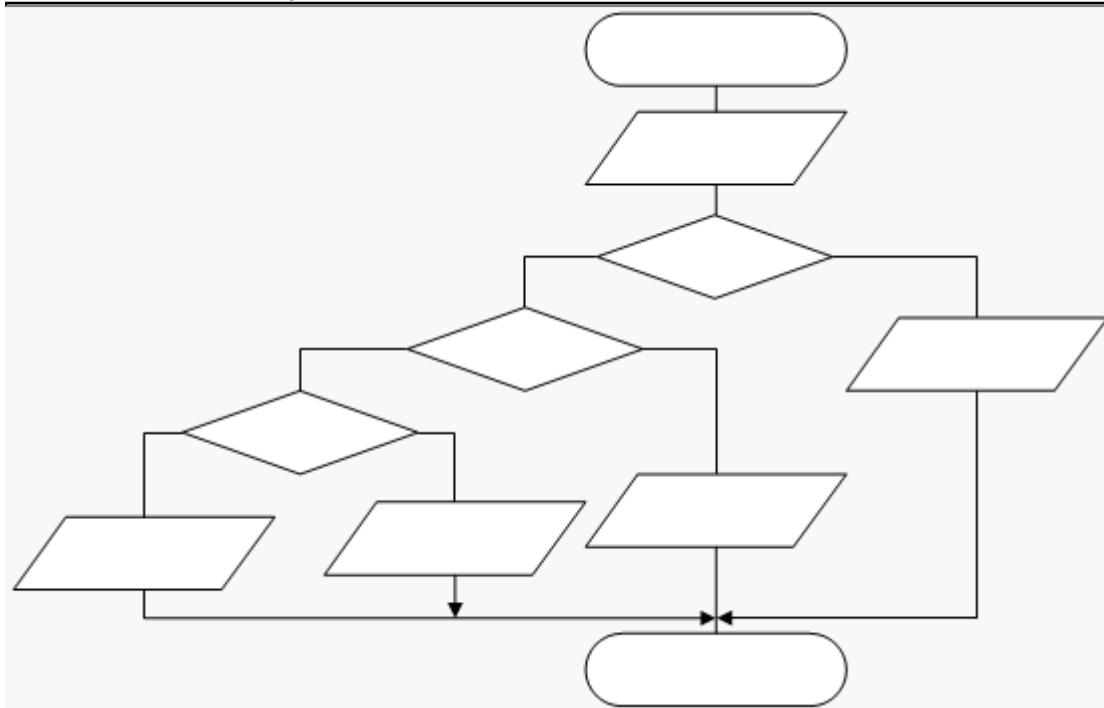
г) a^{12} за четыре операции.

10. По заданной блок-схеме алгоритма восстановите условие задачи.

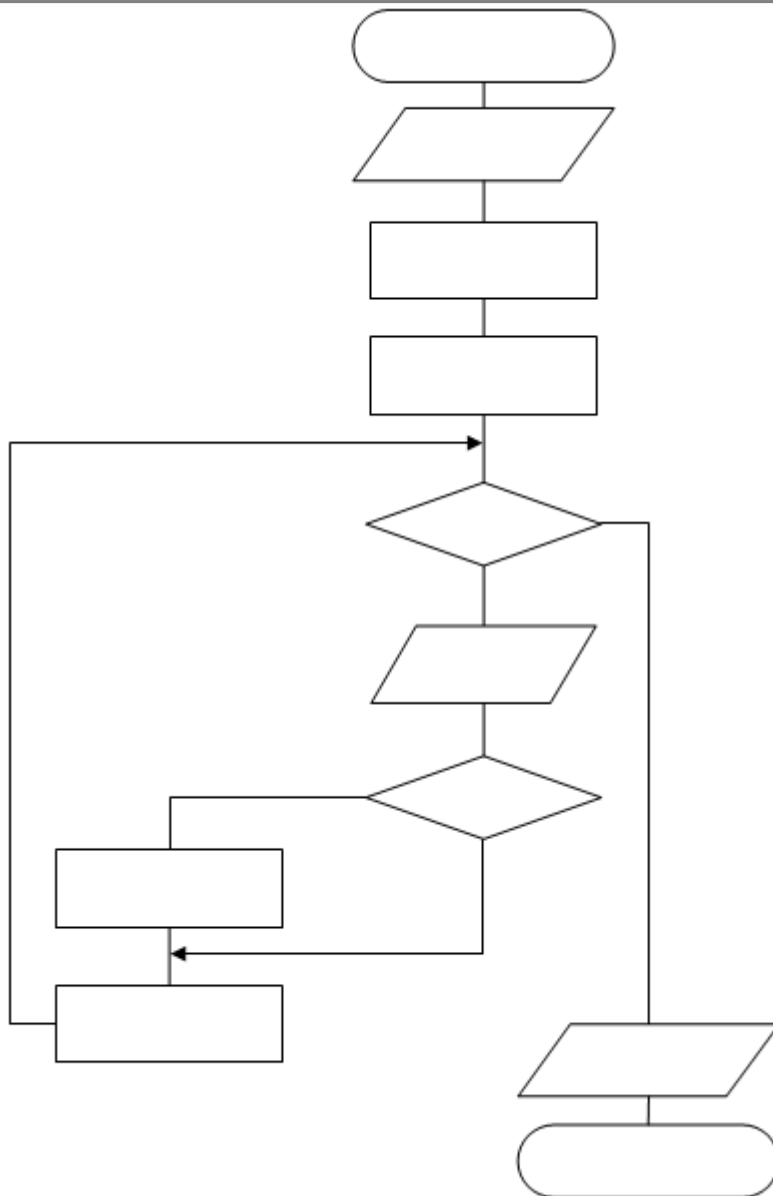


шаг алгоритма	аргумент			промежуточные величины		результат	пояснения
	r1	r2	pi	s1	s2	S	
1							
2							
3							
4							
5							
6							

11. Артериальное кровяное давление у студентов колледжа (16-20 лет) считается нормальным, если верхнее артериальное давление $H < 120$, а нижнее $h > 70$ мм рт.ст. Причем $H - h > 30$. Составьте алгоритм, который в зависимости от измеренных значений H и h выдает одно из следующих значений: «нормальное», «повышенное», «пониженное».



12. Составьте блок-схему алгоритма нахождения максимального числа среди заданных n чисел ($n > 2$).



Задача №2

Составьте программу для выполнения расчетов функции. Значение аргумента должно задаваться в диалоге с пользователем.

Пример варианта:

$$A = \left(x + \frac{7}{6}\right)^{\frac{4}{3}} + \sin e^x + \arcsin(\cos 2x)$$

Задача №3

Вычислите заданное целочисленное выражение для данных a, b. Результат X - тоже целочисленный. Исходные данные и результат должны проверяться на область допустимых значений.

Пример варианта:

$$X = \begin{cases} a / b + 31, & \text{если } a > b, \\ -25, & \text{если } a = b, \\ (a * 5 - 1) / a, & \text{если } a < b; \end{cases}$$

Задача №4

Решите задачу с использованием оператора switch.

Пример варианта:

Единицы длины пронумерованы следующим образом: 1 – дециметр, 2 – километр, 3 – метр, 4 – миллиметр, 5 – сантиметр. Дан номер единицы длины и длина отрезка L в этих единицах (вещественное число). Вывести длину данного отрезка в метрах.

Задача №5

Осуществить ввод необходимых данных, выполнить реализацию алгоритма с использованием оператора цикла while, обеспечить вывод полученных результатов. Не разрешается использовать другие операторы цикла. Не допускается использование массивов и операторов, прерывающих ход программы (break, goto).

Пример варианта:

Определить, является ли заданное число совершенным, т.е. равным сумме всех своих (положительных) делителей, кроме самого этого числа (например, число 6 совершенно: $6 = 1 + 2 + 3$). Найти все совершенные числа в диапазоне [a, b].

Задача №6

Выполните задание согласно варианту.

Пример варианта:

1. Расчет конечных сумм.

Составить программу расчета конечной суммы. Число членов суммы вводится с клавиатуры. Получить 3 значения при различных N. Результат сравнить с контрольным значением. Вычислить погрешность.

Вид суммы: $2 + 4 + 6 + 8 + \dots + 2n$

Контрольное значение: $n(n+1)$

2. Разложение функций в ряд Маклорена (Тейлора).

Составить программу, вычисляющую $f(x)$ в виде бесконечного ряда с заданной точностью $\varepsilon = 10^{-3}$ (10^{-6} , 10^{-9}). Значение x вводится с клавиатуры. Проверить вычисления с помощью системной функции. Подсчитать количество итераций, потребовавшихся для получения результата с заданной точностью.

Функция $f(x)$: $\sin x$

$$x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots$$

Разложение в ряд:

Собеседование, опрос/Контрольная работа №1

- Определение алгоритма.
- Свойства алгоритма.
- Данные в алгоритмах.
- Формы представления алгоритмов.
- Правила построения блок-схем.
- Определение программы.
- Этапы решения задач.
- Критерии оценки алгоритма
- Базовые алгоритмические структуры.
- Дополнительные структуры. Реализация с помощью базовых структур.
- Разветвленная структура. Ветвление, полная и неполная формы.

- Определение цикла. Виды циклов.
- Определение языка программирования.
- Подготовка исполнимого файла программы.
- Жизненный цикл ПП.
- Интерфейс среды разработки.
- Средства отладки.
- Структура программы на языке C/C++.
- Препроцессор.
- Типы данных.
- Константы.
- Переменные. Определение. Идентификаторы.
- Дайте определения следующих терминов: а) вводные комментарии, б) оглавление, в) пояснительные комментарии г) отступы.
- Почему программы должны быть удобочитаемыми.
- Операции. Понятие операнда. Унарные, бинарные, тернарные операции.
- Выражения. Приоритет операций.
- Неявные преобразования типов. Операция приведения типов.
- Ввод-вывод в языке C. Стандартные потоки. Функции ввода-вывода.
- Ввод-вывод в языке C++. Стандартные потоки. Операции ввода-вывода.
- Какие алгебраические выражения соответствуют следующим записям на языке C++: а) $(a + b) / c$; б) $a + b / c$; в) $a / b / c$; г) $a / (b * c)$; д) $(a + b) / (d + c)$; е) $a + b / (d + c)$; ё) $a + b / d + c$; ж) $(a + b) / d + c$.
- Что называется оператором.
- Операторы управления работой программы. Группы операторов управления.
- Составной оператор. Отличие от блока.
- Условный оператор. Полная и сокращенная форма. Вложенные условные операторы.
- Переключатель. Реализация с помощью переключателя конструкции выбора.
- Циклы с предусловием и постусловием.
- Цикл с параметром.
- Операторы перехода.

Доклад, сообщение/Реферат №1

- Средства и языки описания (представления) алгоритмов.
- Современные парадигмы программирования.
- Структурный подход к программированию.
- Объектно-ориентированный подход к программированию.
- Декларативный подход к программированию.
- Параллельное программирование.
- Case-технологии разработки программных систем.
- Доказательное программирование.
- Системы программирования.
- О фирмах-разработчиках систем программирования.
- Языки программирования. Характеристики языков программирования.
- Языки программирования C-семейства.
- Проблема алгоритмической разрешимости в математике.
- Жизненный цикл программных систем.
- Методы управления проектами при разработке программных систем.

Расчетно-графическая работа №1

Пример варианта:

1. Дано целое N ($9 < N < 1000000$). Разработайте алгоритм и напишите часть кода программы для ответа на вопросы, верно ли, что:

$y = \frac{1 - x^2}{1 + x^4}$	-3	3	2	-0,176	Таблица значений функции: +-----+ X Функция +-----+ X = ... Y = ... X = ... Y = +-----+ Составил: < ФИО >
-------------------------------	----	---	---	--------	---

Собеседование, опрос/Контрольная работа №2

- Объявление одномерных и многомерных массивов.
- Размещение в памяти одномерных и многомерных массивов.
- Обращение к элементу массива.
- Инициализация массива.
- Ввод/вывод элементов одномерного и двумерного массивов.
- Алгоритм поиска максимального/минимального элементов массива.
- Понятие указателя.
- Операции с указателями.
- Передача параметров в функцию через указатель.
- Динамические массивы.
- Определение функции, объявление (прототип функции).
- Формальные и фактические параметры функции.
- Передача функции параметров по значению и по адресу. Передача в качестве параметров массивов.
- Если явно не указан тип возвращаемого значения, какой тип возвращает функция по умолчанию?
- Как функция может вернуть значение? Сколько значений может вернуть функция?
- Что такое тип функции?
- Области видимости.
- Что обозначает фраза: "переменная скрывает другую переменную"?
- Классы памяти и области видимости переменных.
- Понятие рекурсивной функции. Когда можно использовать рекурсивную функцию? Можно ли заменить использование рекурсивной функции другими операторами, если можно, то как?
- Аргументы функции main.
- Функции с переменным числом параметров.
- Указатели на функции. Использование указателей на функции в качестве параметров функций.
- Как объявить массив указателей на функции. Как осуществить вызов функций-элементов массива?
- Что такое код символа?
- Что такое таблица ASCII-кодов?
- Какой тип данных используется для представления кода символа?
- Функции стандартной библиотеки для определения класса символов.
- Функции стандартной библиотеки для преобразования символов.
- Представление строк в С.
- Взаимоотношения между строкой и массивом символов.
- Основные операции со строками.
- Что такое препроцессор?
- Что такое код программы в препроцессированной форме?
- Какие функции выполняет препроцессор?

- Что такое директива препроцессора?
- Заранее определенные макросы (псевдопеременные).
- Что такое заголовочный файл? Какую информацию обычно содержит заголовочный файл? Мероприятия по предотвращению повторного включения файлов.
- Включение файлов.
- Условная компиляция.
- Макроопределения.
- Определение структурного типа.
- Чем отличаются записи:

```
struct A {int x, y};  
struct {int x, y;} A;  
struct B {int x, y;} A;  
typedef struct {int x, y;} A;  
typedef struct B {int x, y;} A;
```
- Как вычислить количество памяти, необходимое для переменной структурного типа?
- Описание переменных структурного типа. Массивы структур, массивы указателей на структуры.
- Операции доступа к элементам структуры.
- Как осуществляется копирование структур, сравнение структур?
- Битовые поля структур.
- Объединения.
- Потоки. Буферизованные потоки. Действия с потоками.
- Стандартные потоки.
- Открытие и закрытие файловых потоков.
- Операции чтения/записи в файловый поток символов, строк, форматированных данных.
- Операции чтения/записи в файловый поток блоков данных.
- Позиционирование в потоке.
- Понятие модуля.
- Нисходящее проектирование/программирование.
- Модульное программирование.
- Библиотеки функций.
- Библиотеки динамической компоновки.

Собеседование, опрос/Контрольная работа №3

Пример варианта:

1. Дан массив целых чисел $x[10]$. Найдите сумму элементов массива, больших числа A (значение A вводится с клавиатуры).
2. В массиве хранится информация о среднесуточной температуре за каждый день месяца. Определите дату самого тёплого дня. Решение оформите в виде отдельной функции. Выбор месяца (и, соответственно, количество элементов массива) выполняет пользователь. Значения элементов массива можно заполнить случайным образом или предложить ввести пользователю.

Расчетно-графическая работа №2

Пример варианта:

1. Заданы матрицы A , B , C максимальной размерности (20, 20). Вычислить матрицу $D = A^T * (C^T + B) * C$. Предусмотреть возможность ввода матриц A , B , C - меньшей размерности. Ввод размерности матриц организовать с терминала. Ввод матриц организовать из файла. В программе предусмотреть проверку размерности исходных матриц на возможность вычисления матрицы D . Результат вычислений вывести в файл.
2. В данном тексте (файл text.txt) уберите лишние пробелы и расставьте правильно пробелы возле знаков препинания ('!', '!', '!', '?').

Задача №13

Пример варианта:

Напишите функцию проверки правильности расстановки скобок в выражении (скобки могут быть круглые, фигурные, квадратные и треугольные).

Задача №14

Пример варианта:

Напишите функцию, удаляющую из линейного однонаправленного списка все вхождения элемента E.

Задача №15

Пример варианта:

Составить описание класса для представления рациональных чисел. Предусмотреть методы для проверки, равны ли друг другу два рациональных числа; сложения рациональных чисел; умножения рациональных чисел. Написать программу, демонстрирующую работу с этим классом. Программа должна содержать меню, позволяющее осуществить проверку всех методов класса.

Задача №16

Пример варианта:

Для класса, разработанного в предыдущей задаче, добавить реализацию методов для класса, выполнив перегрузку операций: «+», «-», «/», «*» – действия с дробями; «*» – умножение дроби на число; «++», «--» – увеличение/уменьшение дроби на единицу.

Задача №17

Разработать иерархию классов. Определить в базовом классе чистую виртуальную функцию Show(), предназначенную для вывода на экран информации о текущем объекте. Для проверки определить массив ссылок на абстрактный класс, которым присваиваются адреса различных объектов.

Пример варианта:

Создать абстрактный класс с виртуальной функцией: норма. Создать производные классы: «комплексное число», «вектор» (из 10 элементов), «матрица» (2x2). Определить функцию нормы: для комплексных чисел – модуль в квадрате, для вектора – корень квадратный из суммы элементов по модулю, для матрицы – максимальное значение по модулю.

Задача №18

Пример варианта:

Создать шаблон контейнерного класса Vect (одномерный динамический массив). Реализовать методы: At(int i) – получение элемента с индексом i; Size() – количество элементов; AddEnd(T t) – добавление элемента t в конец вектора; GetMaxSize() – получение максимального размера вектора; GetMaxSize(int i) – установка максимального размера вектора. Разработать метод Show() – вывод на экран всех элементов контейнерного класса. Показать инстанцирование шаблона для типов int, double, Point (класс «Точка на плоскости» разработать самостоятельно; предусмотреть необходимые методы для решения основной задачи – например, перегрузку операции сравнения).

Задача №19

Пример варианта:

Напишите программу, формирующую по заданному вектору целых чисел список из элементов вектора с четными числами.

Собеседование, опрос/Контрольная работа №4

- Понятие структуры данных.
- Связные списки. Разновидности списков.
- Операции с линейными однонаправленными списками.
- Линейный двунаправленный список.
- Циклические списки.
- Ортогональные списки.
- Стеки в вычислительных системах.
- Реализация стека с помощью связанного списка.

- Логическая структура очереди.
- Реализация очереди с помощью связного списка.
- Логическое представление и изображение деревьев.
- Обходы дерева.
- Частные случаи деревьев.
- Общая характеристика алгоритмов сортировки данных. Классы методов сортировки.
- Определение сложности алгоритмов.
- Внешняя сортировка.
- Последовательный и бинарный поиск в массиве.
- Классы как механизм создания новых типов.
- Поля и методы класса.
- Управление доступом.
- Конструкторы и деструктор.
- Конструктор копирования.
- Пространства имен.
- Статические члены класса.
- Константные члены класса.
- Перегрузка унарных операторов.
- Перегрузка бинарных операторов.
- Друзья класса.
- Перегрузка операций извлечения и вставки.
- Наследование.
- Виртуальные функции.
- Абстрактные функции.
- Информация о типе на этапе выполнения. RTTI.
- Множественное наследование.
- Потоки ввода-вывода. Ввод и вывод типов, определяемых пользователем.
- Шаблоны классов. Определение шаблона. Инстанцирование. Параметры шаблонов и проверка типов.
- Шаблоны функций.
- Специализация.
- Наследование и шаблоны.
- Обработка исключений.
- Строки в C++.
- Библиотека стандартных шаблонов. Контейнеры, итераторы, объекты-функции.
- Библиотека стандартных шаблонов. Виды контейнеров, последовательные и ассоциативные контейнеры, адаптеры.
- Библиотека стандартных шаблонов. Итератор как обобщение указателя, классы итераторов.
- Библиотека стандартных шаблонов. Алгоритмы.

Расчетно-графическая работа №3

1. Создайте базовый класс «фигура», и производные классы: «круг», «прямоугольник», «трапеция».
2. Определите перегруженные операции извлечения и вставки для объектов данных типов.
3. Создайте случайным образом (или в диалоге с пользователем) набор фигур (круги, прямоугольники, трапеции), указатели на которые сохраните в массиве указателей на класс «фигура».
4. Выведите в файл out.txt информацию о созданных объектах.
5. Вычислите общую площадь, занимаемую созданными объектами.

9.2. Примерный перечень тем курсовой работы

Не предусмотрено учебным планом

9.3. Типовые контрольные задания для промежуточной аттестации: зачет, экзамен

Примерный перечень теоретических вопросов к зачету №1

- Понятие алгоритма и его основные свойства.
- Характеристика основных этапов разработки алгоритмов.
- Основные формы и способы представления алгоритмов.
- Основные этапы решения задач на ЭВМ.
- Структура системы программирования. Последовательность обработки программы от исходного текста на языке программирования до исполняемого кода.
- Структура алгоритмического языка. Понятие синтаксиса, семантики, лексики.
- Основные лексические структуры языка. Служебные слова. Идентификаторы. Переменные и константы.
- Целочисленные и вещественные константы. Запись числовых констант в различных системах счисления. Представление вещественных констант в десятичной и экспоненциальной формах.
- Символьные константы. Управляющие последовательности. Строковые константы.
- Препроцессорные преобразования программы. Директивы препроцессора.
- Классификация типов данных языка C++. Определение и инициализация переменных в языке C++. Идентификаторы и модификаторы базовых типов. Символьные переменные. Тип void.
- Выражения и операции. Правила записи выражений. Приоритет и порядок выполнения операций. Приведение типов. Правила преобразования типов. Операция явного приведения типа.
- Генерирование случайных чисел в заданном промежутке.
- Основные принципы структурного программирования. Теорема Бёма-Якопини о структурировании программ.
- Базовые управляющие структуры: следование, ветвление, цикл. Общая характеристика и классификация операторов.
- Условный оператор и переключатель. Использование вложенных условных операторов.
- Операторы циклов. Взаимозаменяемость операторов циклов.
- Операторы перехода break и continue. Оператор безусловного перехода и ограничения в его использовании.
- Составные операторы.
- Особенности организации операций ввода-вывода в языке C++. Заголовочные файлы и их структура. Стандартная библиотека ввода-вывода.

Примерный перечень практических заданий к зачету №1

1. Составьте программу для выполнения расчетов функции. Значение аргумента должно задаваться в диалоге с пользователем.

Пример варианта:

$$A = \left(x + \frac{7}{6}\right)^{\frac{4}{3}} + \sin e^x + \arcsin(\cos 2x)$$

2. Вычислите заданное целочисленное выражение для данных a, b. Результат X - тоже целочисленный. Исходные данные и результат должны проверяться на область допустимых значений.

$$X = \begin{cases} a / b + 31, & \text{если } a > b, \\ -25, & \text{если } a = b, \\ (a * 5 - 1) / a, & \text{если } a < b; \end{cases}$$

3. Решите задачу с использованием оператора switch. Единицы длины пронумерованы следующим образом: 1 – дециметр, 2 – километр, 3 – метр, 4 – миллиметр, 5 – сантиметр. Дан номер единицы длины и длина отрезка L в этих единицах (вещественное число). Вывести длину

данного отрезка в метрах.

4. Осуществить ввод необходимых данных, выполнить реализацию алгоритма с использованием оператора цикла while, обеспечить вывод полученных результатов. Не разрешается использовать другие операторы цикла. Не допускается использование массивов и операторов, прерывающих ход программы (break, goto). Определить, является ли заданное число совершенным, т.е. равным сумме всех своих (положительных) делителей, кроме самого этого числа (например, число 6 совершенно: $6 = 1+2+3$). Найти все совершенные числа в диапазоне [a, b].

5. Составить программу расчета конечной суммы. Число членов суммы вводится с клавиатуры. Получить 3 значения при различных N. Результат сравнить с контрольным значением. Вычислить погрешность.

Вид суммы: $2 + 4 + 6 + 8 + \dots + 2n$

Контрольное значение: $n(n+1)$

6. Составить программу, вычисляющую $f(x)$ в виде бесконечного ряда с заданной точностью $\epsilon=10^{-3}$ (10^{-6} , 10^{-9}). Значение x вводится с клавиатуры. Проверить вычисления с помощью системной функции. Подсчитать количество итераций, потребовавшихся для получения результата с заданной точностью.

Функция $f(x)$: $\sin x$

$$x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots$$

Разложение в ряд:

7. Найти все натуральные числа, не превосходящие заданного m , десятичная запись которых есть строго убывающая последовательность цифр.

8. Вводится натуральное число n . Определить, является ли оно степенью двойки.

9. Проверить, содержит ли квадрат данного натурального числа n цифру 3 в своей записи. Найти все двузначные такие числа.

10. Найти все натуральные числа, не превосходящие заданного m , десятичная запись которых есть строго убывающая последовательность цифр.

Примерный перечень теоретических вопросов к зачету №2

- Массивы Определение и инициализация элементов одномерного массива. Основные алгоритмы обработки одномерных массивов.
- Многомерный массив. Размещение элементов многомерного массива в памяти и организация доступа к ним. Основные алгоритмы обработки двумерных массивов.
- Представление строки как одномерного массива символов. Особенности определения и инициализации строковых переменных. Основные функции для выполнения операций над строками. Функции стандартной библиотеки для обработки символьных массивов.
- Адреса и указатели. Объявление и инициализация указателей. Определение адреса переменной в памяти. Получение значения переменной по указанному адресу. Особенности реализации арифметических операций над указателями.
- Массивы и указатели. Использование указателей при обработке массивов. Динамические массивы. Операции выделения и освобождения динамической памяти.
- Использование механизма указателей для доступа к элементам одномерного массива. Определение массива указателей и инициализация его элементов. Особенности внутреннего представления массива указателей в памяти ЭВМ.
- Структурный тип данных. Объявление структурной переменной и инициализация ее полей. Доступ к элементам структуры. Способы доступа к элементам структуры через указатель на нее.
- Функции. Определение и описание функций. Формальные и фактические параметры. Указатели на функции.
- Функции. Локальные и глобальные переменные. Способы обмена данными между программой и функциями.
- Функции. Рекурсивные функции.

Указатели в параметрах функций. Ссылочный тип, передача параметров по ссылке. Массивы и строки как параметры функций. Перегрузка функций.

- Функции. Указатели на функции. Использование функций в качестве параметров.
- Функции. Рекурсивные функции. Функции с переменным количеством параметров.
- Функции. Функция main. Аргументы функции main.
- Препроцессор. Директивы #include, #define. Директивы условной компиляции.
- Ввод/вывод данных с использованием средств языка C++. Поточные классы. Операторы извлечения и вставки.
- Ввод/вывод данных с использованием средств языка C++. Файловые потоки. Режимы доступа к файлу, режимы открытия файла.
- Форматированный ввод-вывод. Манипуляторы.
- Строки. Объявление и инициализация массива символов (строк). Нулевой символ. Библиотечные функции для работы со строками.
- Структуры. Объявление и инициализация. Доступ к элементам структуры. Указатели на структуры.
- Переименование типов. Перечисления.
- Концепция объектно-ориентированного программирования (инкапсуляция, наследование, полиморфизм). Классы. Различия между class и struct. Поля. Методы для обработки полей. Уровни видимости.
- Методы класса. Классификация. Реализация методов класса. Неявный указатель this.
- Экземпляры класса: статическое и динамическое создание объектов. Массивы объектов.
- Статические члены класса. Константные члены класса.
- Конструкторы и деструктор. Конструктор копирования.
- Друзья класса.
- Операторные функции. Перегрузка унарных операций. Перегрузка бинарных операций. Перегрузка операции присваивания; отличие от конструктора копирования.
- Ссылки. Передача значений по ссылке и через указатели.
- Простое наследование. Правила определения производного класса. Наследование полей и методов. Открытое и закрытое наследование.
- Указатели на базовый и производный классы. Вызов методов по указателю на класс. Статическое и динамическое связывание. Виртуальные функции. Чисто виртуальные функции, абстрактные базовые классы.
- Проектирование иерархий классов. Эффективное имя класса. Использование информации о типе во время выполнения (RTTI – Run-Time Type Information).
- Шаблоны. Инстанцирование. Параметры шаблонов.
- Обработка ошибок. Исключения. Перехват исключений.
- Строки в C и C++
- Связные списки. Типовые задачи обработки линейных однонаправленных/двунаправленных списков: создание, добавление элементов, удаление элементов, печать, поиск элемента с заданными свойствами.
- STL

Примерный перечень практических заданий к экзамену: Вопрос № 2

- Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 — катет a , 2 — гипотенуза c , 3 — высота h , опущенная на гипотенузу, 4 — площадь S . Дан номер одного из этих элементов и его значение. Выведите значения остальных элементов данного треугольника (в том же порядке).
- Напишите программу, печатающую количество цифр в десятичной записи введенного натурального числа.
- Найдите все натуральные числа, не превосходящие заданного m , сумма цифр в десятичном представлении каждого из которых равна заданному k .
- Найдите первые m более чем 2-разрядных чисел-палиндромов, то есть чисел, десятичная запись

которых читается одинаково в прямом и обратном направлениях, например: 353, 234432.

- Найдите все натуральные числа, не превосходящие заданного m , десятичная запись которых есть строго убывающая последовательность цифр.
- Каждое из заданных натуральных чисел замените числом, получающимся при записи его десятичных цифр в обратном порядке.
- Напишите программу разложения на простые множители натурального числа.
- Определите, является ли число n простым.
- Напечатать все простые числа до n . n вводится с клавиатуры.
- Найдите все представления положительного целого числа n в виде произведения двух чисел.
- Определите, чередуются ли положительные и отрицательные элементы в целочисленном массиве из 20 элементов. Если элементы чередуются, то вычислить сумму положительных элементов, иначе вычислить произведение отрицательных элементов.
- Составьте функцию выделения из массива X элементов со значением в диапазоне $X1..X2$. Результат из M элементов помещается в массив Y .
- Дан произвольный текст. Напечатайте в алфавитном порядке все буквы, которые входят в этот текст по одному разу.
- Распечатайте введенную строку, исключив из нее те символы, которые находятся между скобками '(' ')'. Сами скобки не удалять. Если хотя бы одной скобки нет – сообщить об этом.
- Подсчитайте количество слов в строке.
- Напишите программу, которая вводит с клавиатуры два непустых массива целых чисел в диапазоне от нуля до девяти, и, считая эти массивы десятичным представлением двух чисел, печатает их сумму.
- Напишите программу, которая вводит с клавиатуры два непустых неубывающих массива целых чисел, и печатает те и только те элементы, которые встречаются хотя бы в одном из массивов (объединение множеств).
- Напишите программу, которая вводит с клавиатуры два непустых неубывающих массива целых чисел, и печатает те и только те элементы, которые встречаются в обоих массивах (пересечение множеств).
- Напишите программу, вводящую натуральное число R , и печатающую количество точек с целочисленными координатами внутри замкнутого шара радиуса R с центром в начале координат.
- Объедините два упорядоченных по возрастанию массива a и b в массив c , упорядоченный по возрастанию.
- Найдите наименьшее общее кратное для всех элементов массива - минимальное число, которое делится на все элементы массива без остатка.
- В строке находится вещественное число (целая часть, символ "точка" и символы-цифры дробной части числа). Требуется преобразовать строку во внутреннюю форму представления (переменную типа `double`).
- В одномерном числовом массиве переместите нулевые элементы в конец, сдвинув остальные элементы влево.
- В двумерном числовом массиве замените нулями все элементы, расположенные на главной диагонали и выше ее.
- В квадратной матрице поменяйте местами столбец, в котором расположен первый нулевой элемент, со столбцом, содержащим последний отрицательный элемент.
- Дана целочисленная прямоугольная матрица. Определите количество столбцов, не содержащих ни одного нулевого элемента.
- Составьте программу, которая читает текстовый файл и определяет, сколько слов начинается на букву 'a'.
- Разработайте программу, которая проверяла бы орфографию (правильность написания) слов в некоторой строке, сравнивая их со словами из словаря. Использовать при этом в качестве словаря одномерный массив слов.

-
- Задан одномерный массив, каждым элементом которого является строка символов, состоящая из одних цифр. Требуется упорядочить элементы массива по возрастанию их числовых значений.
 - Опишите структуру с именем HOTEL, содержащую следующие поля: название отеля; тип номера; стоимость номера. Напишите программу, выполняющую следующие действия: ввод с клавиатуры (или из файла) данных в массив, состоящий из девяти элементов типа HOTEL; вывод на экран содержимого массива в алфавитном порядке по названиям отелей; вывод на экран номера с наибольшей стоимостью с названием отеля и указанием типа (Если их несколько, вывести весь список).
 - Реализовать тип данных «МАТРИЦА», содержащий
переменные-члены: количество строк; количество столбцов; указатель на массив элементов
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; получение элемента под номером (i,j); получение и изменение размерности матрицы; сложение, вычитание и умножение матрицы на число
 - Реализовать тип данных «МАТРИЦА», содержащий
переменные-члены: количество строк; количество столбцов; указатель на массив элементов
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; сложение и вычитание двух матриц; транспонирование матрицы; проверка равенства размерности двух матриц
 - Реализовать тип данных «МАТРИЦА», содержащий
переменные-члены: количество строк; количество столбцов; указатель на массив элементов
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор
перегруженные операторы: сложение, вычитание и умножение двух матриц - операторы «+», «-»; копирование матрицы - оператор «=»; ввод с экрана и вывод на экран - операторы «<<>» и «>>»
 - Реализовать тип данных «ПРЯМОУГОЛЬНИК», содержащий
переменные-члены: координаты центра прямоугольника; ширина и высота прямоугольника
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; перенос прямоугольника на расстояние dx, dy; масштабирование прямоугольника (умножение размеров на константу)
перегруженные операторы: проверка двух прямоугольников на равенство - операторы «==» и «!=»
 - Реализовать тип данных «ПРЯМОУГОЛЬНИК», содержащий
переменные-члены: координаты центра прямоугольника; ширина и высота прямоугольника
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; проверка попадания точки в прямоугольник
перегруженные операторы: масштабирование прямоугольника - оператор «*» (умножение размеров на константу); ввод с экрана и вывод на экран - операторы «<<>» и «>>»
 - Реализовать тип данных «ТОЧКА», содержащий
переменные-члены: координаты точки
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; проверка двух точек на равенство
перегруженные операторы: сложение и вычитание двух точек - операторы «+», «-»; ввод с экрана и вывод на экран - операторы «<<>» и «>>»
 - Реализовать тип данных «КРУГ», содержащий
переменные-члены: координаты центра; радиус
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; получение площади и периметра; проверка попадания точки в круг
перегруженные операторы: ввод с экрана и вывод на экран - операторы «<<>» и «>>»
 - Реализовать тип данных «КРУГ», содержащий
переменные-члены: координаты центра круга; радиус
функции-члены: конструкторы (по умолчанию, полный, копирования) и деструктор; получение вершин описанного квадрата

перегруженные операторы: масштабирование круга - оператор «*»; ввод с экрана и вывод на экран - операторы «<<» и «>>»

- Напишите функцию, которая должна перенести в конец линейного однонаправленного списка его первый элемент.

- Напишите функцию, которая в линейном двунаправленном списке удваивает каждое вхождение элемента E.

Раздел билета	Компетенции	Планируемые результаты обучения по дисциплине	Количество баллов
Вопрос №1 Теоретический вопрос (проверяет знания («знать»), сформированные дисциплиной)	ОПК-2 ОПК-3 ОПК-4 ОПК-5 ОПК-7	Знать назначение стандартных инструментальных средств разработки программного обеспечения. Знать общие принципы построения алгоритмов, основные алгоритмические конструкции. Знать базовые понятия и элементы языка программирования C++. Знать основные правила форматирования кода на языке программирования C++. Знать современные программные средства для практической реализации программного обеспечения на персональных компьютерах. Знать основы структурного программирования на языке C++. Знать основы объектно-ориентированного программирования на языке C++.	45

09.03.03 Прикладная информатика, направленность (профиль) "Прикладная информатика в экономике"
 Рабочая программа дисциплины
 Дисциплина: Б1.О.22 Алгоритмизация и программирование
 Форма обучения: очная, заочная
 Разработана для приема 2021/2022, 2022/2023 учебного года
 Обновлено на 2023/2024 учебный год

Раздел билета	Компетенции	Планируемые результаты обучения по дисциплине	Количество баллов
Вопрос №2 Практическое задание (проверяет умения («уметь»), проверяет практические навыки («владеть»), сформированные дисциплиной)	ОПК-2 ОПК-3 ОПК-4 ОПК-5 ОПК-7	<p>Знать назначение стандартных инструментальных средств разработки программного обеспечения.</p> <p>Уметь выбирать современные информационные технологии и программные средства при решении задач профессиональной деятельности. Уметь находить и анализировать техническую документацию по использованию инструментальных средств разработки программного обеспечения.</p> <p>Владеть навыками использования современных инструментальных средств при программировании на языке C++.</p> <p>Знать общие принципы построения алгоритмов, основные алгоритмические конструкции. Знать базовые понятия и элементы языка программирования C++.</p> <p>Уметь реализовывать алгоритмы решения задач на языке программирования C++.</p> <p>Владеть навыками чтения и понимания программ, реализующих несложные алгоритмы обработки числовых и текстовых данных. Владеть навыками подготовки обзоров, докладов с учетом знаний в области алгоритмизации и программирования.</p> <p>Знать основные правила форматирования кода на языке программирования C++.</p> <p>Уметь использовать комментарии в программном коде как способ первичного документирования программы.</p> <p>Владеть навыками форматирования кода и его комментирования на языке программирования C++.</p> <p>Знать современные программные средства для практической реализации программного обеспечения на персональных компьютерах.</p> <p>Уметь осуществлять инсталляцию, настройку инструментальных средств разработки программного обеспечения для решения задач на языке C++.</p> <p>Владеть навыками инсталляции инструментальных средств разработки программного обеспечения для решения задач на языке C++.</p> <p>Знать основы структурного программирования на языке C++. Знать основы объектно-ориентированного программирования на языке C++.</p> <p>Уметь создавать программы на языке C++ для решения типовых задач базового уровня из различных предметных областей.</p> <p>Владеть практическими навыками составления алгоритмов для решения задач и их реализации на языке программирования C++. навыками тестирования и отладки программ.</p>	55