

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Забелина Светлана Алексеевна
Должность: Ректор
Дата подписания: 21.01.2024 19:34:30
Уникальный программный ключ:
ac61efa1186e39eefc0a742ef4d821f52734a481

**Автономная некоммерческая организация высшего образования
«Московский информационно – технологический университет – Московский
архитектурно– строительный институт»**

**ПРОГРАММА
ВСТУПИТЕЛЬНОГО ИСПЫТАНИЯ ДЛЯ ПОСТУПАЮЩИХ
09.04.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА**

Содержание

1. Общие положения	3
2. Требования к уровню подготовки поступающего	3
3. Содержание программы	4
4. Вопросы вступительного испытания по междисциплинарному экзамену	17
5. Порядок, форма и язык проведения вступительного испытания.....	24
6. Продолжительность вступительного испытания.....	25
7. Шкала оценивания	25
8. Список литературы	26

1. Общие положения

1.1. Программа вступительного испытания по междисциплинарному экзамену предназначена для лиц, имеющих право поступать на обучение по программам магистратуры.

1.2. Целью вступительного испытания является установление уровня знаний абитуриента по направлению 09.04.01 «Информатика и вычислительная техника» в объеме образовательной программы высшей школы исходя из Федерального компонента государственного высшего образования. Содержание вступительных испытаний направлено на выявление имеющихся знаний и умений по специальности.

2. Требования к уровню подготовки поступающего

Абитуриент должен

знать:

- основных стандартов оформления технической документации на различных стадиях жизненного цикла информационной системы;
- основ системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем;
- принципов формирования и структуры бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием;
- методов настройки, наладки программно-аппаратных комплексов;
- алгоритмических языков программирования, операционных систем и оболочек, современных сред разработки программного обеспечения;
- классификации программных средств и возможностей их применения для решения практических задач.

уметь:

- решать стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования;
- выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности;
- решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;
- применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы;
- выполнять параметрическую настройку информационных и автоматизированных систем;
- анализировать цели и ресурсы организации, разрабатывать бизнес планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием;
- анализировать техническую документацию, производить настройку, наладку и тестирование программно- аппаратных комплексов;
- составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули
- находить и анализировать техническую документацию по использованию программного средства, выбирать и использовать необходимые функции программных средств для решения конкретной задачи.

3. Содержание программы

Тема 1. Введение в язык C++

Язык программирования C++, характеристика, назначение. Классификация языков программирования: машинный язык, язык ассемблера, языки высокого уровня. Понятие программы, исходная, объектная и исполняемая программы. Трансляторы: ассемблеры и компиляторы. Элементы языка C++: алфавит, константы, идентификаторы, ключевые слова, комментарии. Основные этапы разработки программы. Принципы структурного программирования. Возможности, предоставляемые интегрированной средой разработки программ (IDE) – Visual Studio. Директивы препроцессора: включение файлов (`#include`) и макроподстановки (`#define`). Базовые (простые) типы данных C++. Приведение типов данных. Определение (создание) переменных различных типов. Константы (литералы): десятичные, шестнадцатеричные, символьные, с плавающей точкой, строковые. Ввод / вывод содержимого переменных на консоль. Консольный ввод / вывод с использованием библиотеки ввода-вывода (функции `printf` и `scanf`) и высокоуровневый ввод-вывод с использованием объектов `cin` и `cout`. Операции арифметические, отношения, логические, побитовые. Алгоритмические конструкции ветвления и цикла на примере операторов `if` и `while`. Отладка программ в интегрированной среде Visual Studio: точки останова, просмотр промежуточного содержимого переменных программы, пошаговый режим отладки. Область видимости и время жизни переменных, понятие блока. Понятие функции в языке C++. Правила работы с функциями: прототип, определение, вызов. Создание простейших функций. Главная функция. Передача аргументов главной функции

Тема 2. Массивы, указатели и ссылки

Понятие массива, возможности, определяемые этим типом данных, при обработке информации. Синтаксис объявления массива в языке C++. Инициализация массива при его создании. Понятие индекса - последовательного номера элемента массива. Доступ к элементам массива через операцию индексации. Многомерные массивы и их размещение в памяти компьютера. Структурная и бесструктурная инициализация многомерных массивов.

Назначение типа данных указатель. Синтаксис объявления переменной типа указатель. Использование операции адресации для инициализации указателя. Доступ к содержимому переменной, адрес которой хранится в указателе. Операции над указателями: присваивание адреса, доступ по указателю (разыменование), увеличение/уменьшение указателя, сравнение указателей, индексация указателей. Связь массивов и указателей, имя массива – указатель на его первый элемент. Массивы указателей. Указатели и динамическое распределение памяти, операторы new и delete. Тип данных ссылка. Создание переменной типа ссылка. Три способа использования ссылки: - передача, как параметра функции, - ссылка, в качестве возвращаемого значения функции, - независимая ссылка.

Тема 3. Операторы управления потоком

Оператор - единица выполнения программы. Простые и составные операторы (блоки). Условный оператор if, синтаксис и логика работы. Вложенные операторы if. Выбор одного из двух - конструкция if – else. Множественный выбор – конструкция if – else if - ... - else. Организация множественного выбора при помощи оператора switch. Синтаксис конструкции switch и порядок выполнения операторов. Операторы организации циклов. Оператор цикла for. Заголовок for, логика работы оператора. Логика работы оператора for при отсутствии любого или всех выражений в его заголовке. Операторы цикла с предусловием while и постусловием do-while, синтаксис и логика работы. Варианты предпочтительного использования операторов цикла в зависимости от решаемой задачи. Вложенные циклы. Операторы прерывания блока: break, continue, return. Использование этих операторов для досрочного завершения циклов. Оператор безусловного перехода goto.

Тема 4. Символьные массивы и строки

Определение строки символов. Способы инициализации строк. Представление строки в памяти, символ конца строки. Строковые литералы, обработка компилятором строкового литерала. Использование признака завершения строки для определения длины строки, копирования, присваивания,

объединения строк. Считывание строк с клавиатуры. Библиотечные функции для работы со строками, примеры использования функций для обработки строк. Массивы строк. Хранения символьных строк различного размера в массиве указателей.

Тема 5. Типы, задаваемые пользователем

Структуры: назначение, шаблон и объявление. Создание структурных переменных. Инициализация структур. Доступ к элементам структуры через операцию точка и через указатель. Определение в качестве элементов структуры битовых полей. Передача структурных переменных, как параметров функциям. Объединения, синтаксис, специфика использования подобных объектов. Доступ к элементам объединения. Перечисляемый тип данных, определение и использование перечисляемого типа. Переименование типов – инструкция typedef.

Тема 6. Функции C/C++

Выделение группы инструкций из основной программы в функцию. Синтаксис создания функций в языке C++ - определение функции. Прототип функции. Передача параметров функции и возврат функцией значения. Перегрузка функций. Рекурсивный вызов функции. Механизм вызова функции и возврат из нее через стек. Особенности передачи параметров функциям по значению и по адресу (указателю, ссылке). Передача в функцию массивов. Указатели на функции. Косвенный вызов функции через указатель. Возможности языка C++, поддерживающего указатели на функции (передача функции в качестве аргумента в другие функции, хранение функций в массивах и структурах). Использование указателей на функции в практических примерах.

Тема 7. Классы

Классы, как типы, определяемые пользователем. Синтаксис объявления класса, члены класса, модификаторы доступа к членам класса. Создание объектов класса. Область видимости членов класса. Указатель this. Инкапсуляция в контексте понятия класса. Инициализация объектов, конструкторы класса. Уничтожение объектов при выходе их из области

видимости, деструкторы. Перегрузка конструкторов: конструкторы без параметров, конструкторы с параметрами, конструкторы копии. Конструктор копирования по умолчанию. Особенности использования конструкторов копии. Присваивание объектов. Передача объектов функциям. Объекты, в качестве возвращаемых значений функций. Создание массивов объектов. Использование указателей на объекты. Динамическое размещение объектов в памяти.

Тема 8. Перегрузка операций

Перегрузка операций как форма полиморфизма. Назначение перегрузки операций. Синтаксис функций-членов класса, осуществляющих перегрузку операторов. Перегрузка бинарных операций: арифметических операций, операций отношения, логических операций. Перегрузка унарных операций. Особенность оператора присваивания, случаи необходимости перегрузки оператора присваивания. Перегрузка операции индексации.

Тема 9. Шаблоны и библиотека стандартных шаблонов

Функция шаблон (родовая функция), назначение. Определение функции-шаблона, примеры. Различие родových и перегруженных функций. Родовые классы – шаблоны классов. Объявление родового класса. Случаи применения шаблонов класса. Создание различных шаблонов классов. Определение понятия контейнер, типы контейнеров (классы-контейнеры). Наиболее востребованные классы-контейнеры: векторы, списки, ассоциативные списки (назначение и возможности, наиболее важные функции-члены, доступ к элементам контейнеров, создание объектов). Строковый класс и возможности работы со строками.

Тема 10. Основы языка C#

Типы данных C#: типы значений и ссылочные типы, концептуальная разница этих типов. Создание и инициализация переменных типа значений (простые типы) и ссылочных типов (массивов и строк). Предопределенные (встроенные) типы значений. Встроенные ссылочные типы: object и string, краткая характеристика. Консольный ввод-вывод. Операторы управления потоком исполнения программы: условные операторы, операторы цикла: while,

for, do-while, foreach. Операторы перехода: goto, break, continue, return. Перечисления. Массивы. Организация взаимосвязанных классов (пространства имен и оператор using).

Тема 11. Классы и структуры C#

Объявление класса. Данные-члены и функции члены (конструкторы, методы, операции, свойства, индексы). Создание экземпляров класса. Объявление методов. Доступ к членам класса. Модификаторы доступа к членам класса. Передача параметров в методы по значению и по ссылке. Объявление конструкторов, перегрузка конструкторов. Понятие свойство в языке C#. Определение свойства. Модификаторы доступа для свойств. Свойства, доступные только для чтения и только для записи. Свойства и потеря производительности. Структуры, объявление структуры. Отличия структур от классов. Случаи предпочтительного использования структур. Специфика работы операции new при выделении памяти для структурных объектов. Конструкторы структур, особенность конструктора копирования. Класс Object и метод ToString(). Назначение механизма перегрузки операций. Логика перегрузки операции сложения для класса, реализующего математический вектор. Перегрузка арифметических операций. Особенность в C# перегрузки операций отношения (сравнения). Ограничения на перегрузку операций в C#.

Тема 12. Наследование C#

Сравнение типов наследования – наследование функциональности (реализации) и наследование интерфейса. Сущность наследования функциональности. Примеры с использованием наследования функциональности. Конструкторы производных классов. Переопределение методов базового класса - виртуальные методы. Особенность синтаксиса переопределения метода в языке C#. Реализация виртуальными функциями динамического полиморфизма. Наследование интерфейса - абстрактные классы и абстрактные методы. Интерфейсы C#, синтаксис объявления. Сходство и различие абстрактных классов и интерфейсов. Создание и использование интерфейсов.

Тема 13. Делегаты и события

Концепция указателей на функции, передача метода, как параметра другому методу. Назначение делегатов. Объявление и примеры использование делегатов. Групповые делегаты, назначение, особенности синтаксиса. Модель событий C#. Объявление события. Регистрация обработчиков события. Уведомление зарегистрированных обработчиков о наступлении события (генерация события).

Тема 14. Коллекции. Основы разработки интерфейса приложений

Концепция коллекции в C#. Массивы списки – класс ArrayList. Создание экземпляра класса. Сохранение объектов в ArrayList. Добавление и удаление элементов. Методы класса. Класс Stack (стек). Хранение временных элементов. Реализует алгоритм Last In First Out. Помещение и извлечение элементов из стека. Методы класса. Класс Queue (очередь). Принцип действия очереди – First In First Out. Методы класса. Пространство имен System.Windows.Forms. Класс формы (class Form), основные свойства, события и методы. Создание и уничтожение экземпляра формы, внешний вид формы. Стандартные элементы управления (Button, ComboBox, Label, TextBox, ListBox, MenuStrip, DataGridView), основные события и свойства. Многодокументный интерфейс (MDI).

Тема 15. Основные понятия дисциплины базы данных

Понятие базы данных. Понятие информация. Понятие данные Концепция баз данных. Система управления базами данных. Информационные системы.

Тема 16. Банки данных

Введение в банки данных. Понятие банка данных (БнД). Требования к БнД. Компоненты БнД. Пользователи БнД. Администраторы БнД (АБД) и их функции. Преимущества и недостатки БнД. Классификация БД, СУБД и БнД.

Тема 17. Моделирование баз данных

Уровни представления моделей данных. Логический уровень. Физический уровень. Внешний уровень. Этапы проектирования баз данных. Инфологическое моделирование предметной области. Способы описания предметной области. Требования, предъявляемые к инфологической модели. Компоненты

инфологической модели. Объекты и классы объектов. Атрибуты объектов. Типы объектов. Виды связей. Классы членства. Построение модели «сущность-связь» (ERмодели).

Тема 18. Проектирование баз данных

Даталогическое проектирование. Ранние модели данных. Общие понятия даталогического проектирования. Подход к даталогическому проектированию. Факторы, влияющие на проектирование БД. Особенности даталогических моделей. Ранние подходы к организации БД. Особенности СУБД, построенных на основе инвертированных файлов. Иерархические системы. Сетевые системы. Структуры данных. Манипулирование данными. Ограничения целостности. Формализация реляционной модели данных. Общие положения реляционного подхода. Базовые понятия реляционных баз данных. Тип данных. Домен в базах данных. Отношение в базах данных. Атрибут. Схема отношения. Кортеж. Схема базы данных. Первичный ключ. Внешний ключ. Связи в базах данных. Типы связей в базах данных. Фундаментальные свойства отношений. Получение реляционной схемы из ER-модели.

Тема 19. Язык управления данными SQL

Язык SQL. История. Первые разработки. Стандартизация. Вопросы совместимости. Средства определения данных. Типы данных SQL. Операторы создания схемы базы данных. Создание и удаление БД. Создание, удаление и изменения структуры таблицы. Операторы создания, удаления и изменения индексов. Использование представлений. Другие возможности SQL.

Тема 20. Классификация технических средств

Средства компьютерной техники, средства коммуникационной техники и средства организационной техники. Определение технических средств информатизации. Классификация технических средств информатизации: по назначению, по спецификации, по размерам.

Тема 21. Основные конструктивные элементы средств вычислительной техники

Блока питания. Корпуса компьютеров. Системные платы. Структура и стандарты шин ПК. Типы процессоров.

Характеристики запоминающих устройств (ЗУ). Основная и специальная память. Базовая система ввода – вывода (BIOS).

Тема 22. Способы организации ввода-вывода в ЭВМ

Способы организации ввода-вывода: программно-управляемый, по прерываниям, по каналу прямого доступа. Структурные схемы и алгоритмы ввода/вывода. Типы шин: системная шина, шина расширения, шины ввода/вывода. Классификации интерфейсов и интерфейсных схем. Системные контроллеры (мосты и концентраторы). Понятие, типы и характеристики чипсетов. Архитектуры чипсетов: классическая архитектура «Северный мост \ Южный мост», архитектура «Acceleratedhub» и неоклассическая архитектура для процессоров AMDK8. Интерфейсы процессоров: слоты и сокет. Внутренние шины ISA, EISA, PCI, PCIExpress, HT и далее.

Тема 23. Стандартные параллельный и последовательный порты

LPT-порт (интерфейс IEEE 1284). COM-порт (интерфейс RS-232C). Характеристики и параметры, программистская модель, режимы работы и алгоритмы. Последовательные интерфейсы ПУ. Проводные интерфейсы USB и FireWire. Общая характеристика, параметры, особенности применения.

Тема 24. Технические средства систем дистанционной передачи информации

Структура и основные характеристики систем дистанционной передачи информации. Средства оперативной связи. Средства дистанционной передачи информации. Дистанционная передача информации.

Тема 25. Сетевые телекоммуникационные технологии

Определение и понятие телекоммуникационных технологий. Виды телекоммуникационных технологий. Системы сотовой подвижной связи. Спутниковые системы связи. Линии связи. Кодирование и мультиплексирование данных. Обнаружение и коррекция ошибок. Возникновение и распознавание

коллизий. Беспроводная передача данных. Связь нескольких источников и нескольких данных.

Тема 26. Информационно-телекоммуникационные сети и интернет

Локальные сети LAN (Local Area Network) и глобальные сети WAN (Wide Area Network). Классы сетей. Адресация в IP сетях. Протоколы межсетевого взаимодействия.

Тема 27. Программное обеспечение телекоммуникационных технологий

Специальное программное обеспечение. Служба ICQ, служба IRC, служба FTP. Электронная почта. Служба телеконференций. Мессенджеры. Социальные сервисы. Сетевые СУБД. CRM-системы. Облачные технологии.

Тема 28. Технологии защиты информации в телекоммуникационных сетях

Угрозы безопасности информации, их виды. Методы и средства обеспечения безопасности информации. Аппаратные средства защиты информации: шумогенераторы, фильтры сети, сканирующие радиоприемники и др. Программные средства защиты данных: антивирусы, программы-блокираторы и др. Организационно-техническая защита информации. Инженерно-технические СЗИ.

Тема 29. Основы операционных систем

Понятие программная система. Декомпозиция и принцип модульности. Концептуальное представление структуры системы (абстрактный тип данных, интерфейс и состояние объекта). Структура системы с точки зрения реализации (реализация объектов в Windows). Назначение и функции операционных систем. Элементы архитектуры ЭВМ: машинный язык, формат машинной команды, такт работы ЭВМ, команда и микрокоманда, внутренние регистры процессора (общего назначения, сегментные, указатель команд, регистр флагов). Эволюция операционных систем. Классификация операционных систем. Возможные режимы работы операционных систем: пакетный, диалоговый, режим реального времени. Понятие операционной среды. Место операционной системы в

архитектуре вычислительной системе. Ядро и вспомогательные модули операционной системы. Привилегированный и пользовательский режимы работы. Многослойная структура операционной системы. Понятие процесса и ресурса. Основные виды ресурсов, распределяемых операционной системой и возможности их разделения. Классификация ресурсов.

Тема 30. Управление памятью в операционных системах

Функции операционной системы по управлению памятью. Пространства символьных имен, виртуальных адресов, и физических адресов компьютера. Отображение символьных имен на физическую память. Иерархия памяти. Адресное пространство процесса. Статическое и динамическое преобразование адресов. Аппаратная поддержка настройки адресов и защиты памяти. Классификация методов распределения памяти. Распределение памяти фиксированными разделами. Распределение памяти динамическими разделами, управление распределением при помощи таблиц свободных и занятых областей и односвязными списками. Сегментация виртуальной памяти. Структура виртуального адреса сегментного способа организации памяти. Таблица дескрипторов сегментов, формат дескриптора. Схема формирования физического адреса при сегментной организации памяти. Организация защиты памяти. Глобальные и локальные адресные пространства. Свопинг сегментов виртуальной памяти. Дисциплины замещения сегментов. Преимущества и недостатки сегментного способа организации памяти. Страничная виртуальная память. Определение физической и виртуальной страницы памяти. Понятие страничного файла. Структура виртуального адреса при страничной организации памяти. Общая схема отображения виртуального адресного пространства задачи на физическую память. Формат дескриптора таблицы страниц. Механизм защиты страничной памяти. Функции диспетчера памяти. Многоуровневые таблицы страниц. Сегментно-страничный способ организации виртуальной памяти. Глобальные и локальные дескрипторные таблицы. Изолированные адресные пространства и уровни привилегий (защита приложений друг от друга и защита ОС от приложений) Особенности

архитектуры процессора Intel Pentium для поддержки виртуального механизма памяти и обработки прерываний.

Тема 31. Управление вводом – выводом

Функции операционной системы, связанные с управлением внешними устройствами. Основные концепции организации ввода-вывода. Проблемы программирования ввода-вывода в распределенных вычислительных системах (проблема разделения ресурсов и доскональное знание протокола обмена). Управление вводом-выводом – одна из функций операционной системы. Адресная шина и пространства основной памяти и ввода. Подключение устройств к пространствам памяти и ввода-вывода. Адресация устройств ввода-вывода, определение порта ввода-вывода, как адресуемого элемента пространства ввода-вывода. Программная модель внешнего устройства (совокупность типовых регистры для обмена с процессором). Команды процессора ввода-вывода, аккумулятор (регистр EAX). Драйверы внешних устройств – интерфейс между ядром ОС и аппаратурой ввода-вывода. Два основных режима работы драйверов внешних устройств (режимов управления вводом-выводом): режим обмена с опросом готовности, режим обмена с прерываниями. Виртуализация устройств ввода-вывода (спулинг). Функции супервизора ввода-вывода. Прямой доступ к основной памяти, понятие канала или процессора ввода-вывода. Системные таблицы ОС для управления операциями обмена с внешними устройствами и отслеживания состояния ресурсов. Процесс (схема) управления вводом-выводом. Синхронный и асинхронный ввод – вывод.

Тема 32. Организация внешней памяти. Файловые системы

Физическая структура НЖМД. Сектор, дорожка, цилиндр, магнитная головка чтения/записи. Координаты C-H-S. Логическая структура магнитного диска. Преимущества деления дискового пространства на разделы. Первый сектор диска - информация о распределении дискового пространства по разделам. Главная загрузочная запись (Master boot record). Таблица разделов диска (Partition table). Формат строки таблицы разделов. Первичный и

расширенный разделы. Понятие активного раздела. Двухступенчатая загрузка ОС. Возможности, предоставляемые менеджерами загрузки (Boot manager). Формирование таблицы разделов диска с помощью специальных утилит (FDisk, Parttion Magic). Файловая система – способ организации данных современных операционных систем. Основные возможности, предоставляемые файловой системой. Компонент операционной системы - система управления файлами, назначение, возможности, предоставляемые пользователям. Монтируемые файловые системы. Реализации файловой системы FAT. Структура логического диска. Корневой каталог и таблица размещения файлов, ее структура. Определение понятия кластер. Файловая система FAT32, основные возможности и недостатки. Возможности файловой системы HPFS. Файловая система новой технологии NTFS. Структура раздела, главная таблица файлов (MFT). Распределение дискового пространства. Понятия файловый узел, битовая карта. Возможности, предоставляемые NTFS. Разрешения NTFS, ограничения доступа к файлам и каталогам. Проблемы восстановления файловых систем и техника протоколирования транзакций. Избыточные дисковые подсистемы RAID.

Тема 33. Сетевые транспортные средства операционных систем

Функциональные компоненты сетевой операционной системы (управление локальными ресурсами, сетевые средства: серверная часть, локальная часть, транспортные средства) Коммутация пакетов, буферы и очереди, методы продвижения пакетов. Сетевые модели и стеки протоколов. Стек протоколов TCP/IP Коммутируемая технология Ethernet. Мосты и коммутаторы, алгоритм прозрачного моста. Адресация в сетях. Аппаратные (локальные), сетевые (глобальные) и символьные адреса. Отображение IP-адресов на MAC-адреса. Таблицы маршрутизации. Протоколы маршрутизации.

4. Вопросы вступительного испытания по междисциплинарному экзамену

1. Особенности языков программирования: машинного, ассемблера, высокого уровня.
2. Базовые элементы языка C++.
3. Основные этапы разработки программы.
4. Базовые типы данных языка C++.
5. Консольный ввод/вывод.
6. Основные операции языка (арифметические, отношения, логические, побитовые), примеры.
7. Блок-схемы алгоритмических конструкций ветвления и цикла.
8. Область видимости и время жизни переменных, понятие блока.
9. Функции C/C++, определение, прототип, вызов.
10. Понятие массива в языке C/C++, размещение элементов массива в памяти, объявление и инициализация массивов.
11. Доступ к элементам массива, понятие индекса, перебор элементов массива в цикле.
12. Многомерные массивы, структурная и бесструктурная инициализация массивов. Размещение многомерных массивов в памяти.
13. Указатели C/C++, назначение, примеры создания.
14. Операции над указателями.
15. Массивы указателей.
16. Связь между массивами и указателями.
17. Понятие ссылки в языке C/C++, допустимые способы использования.
18. Передача ссылки в качестве параметра в функцию.
19. Ссылка в качестве возвращаемого значения функции.
20. Понятие оператора, простые и составные операторы (блоки).
21. Оператор условия (if), логика работы, примеры.
22. Вложенные операторы if.

23. Множественный выбор с использованием конструкции if-else if -...-else.
24. Множественный выбор, оператор switch.
25. Оператор цикла while, примеры, варианты предпочтительного использования.
26. Оператор цикла for, примеры, варианты предпочтительного использования.
27. Оператор цикла do while, примеры, варианты предпочтительного использования.
28. Вложенные циклы, примеры.
29. Оператор break, назначение, примеры использования.
30. Оператор continue, назначение, пример использования.
31. Операторы return и goto.
32. Строка – определение, инициализация, представление в памяти.
33. Считывание строк с клавиатуры.
34. Обработка строк, при использовании признака конца строки.
35. Библиотечные функции для работы со строками.
36. Массивы строк.
37. Создание структурных переменных.
38. Доступ к элементам структуры.
39. Передача структурных переменных функциям.
40. Объединения.
41. Перечисления.
42. Работа с функциями в языке C/C++: объявление, определение, вызов.
43. Перегрузка функций.
44. Рекурсивный вызов функций.
45. Механизм использования стека при вызове функций и возврате из них.
46. Передача параметров функции по значению и по адресу.
47. Указатели на функции, косвенный вызов функций.
48. Передача функции в качестве аргумента в другие функции.
49. Массивы указателей на функции.

50. Определение класса, данные-члены и функции члены, спецификаторы доступа.
51. Поведение (функциональность) класса.
52. Конструкторы и деструкторы (создание и удаление объектов).
53. Механизм инкапсуляции.
54. Перегрузка конструкторов.
55. Конструктор копирования.
56. Передача объектов функциям, объекты в качестве возвращаемых значений функций, примеры.
57. Массивы объектов.
58. Размещение объектов в динамической памяти.
59. Назначение перегрузки операций, определение функции, реализующей перегрузку.
60. Перегрузка арифметических операций, пример.
61. Перегрузка операций отношения, пример.
62. Перегрузка логических операций, пример.
63. Перегрузка операции индексации, пример.
64. Перегрузка операции присваивания, пример.
65. Концепция наследования в объектно-ориентированном подходе разработки программ.
66. Создание производного класса.
67. Возможности доступа из производного класса к членам базового класса, защищенные члены класса.
68. Конструкторы и наследование, передача аргументов конструктору базового класса.
69. Иерархия наследования.
70. Контейнерные классы.
71. Сравнение механизма наследования с контейнерным классом, пример.
72. Шаблоны в языке C++, назначение.
73. Родовая функция.

74. Родовой класс.
75. Понятие контейнер в языке C++.
76. Библиотека стандартных шаблонов: векторы.
77. Библиотека стандартных шаблонов: списки.
78. Библиотека стандартных шаблонов: ассоциативные списки.
79. Строковой класс в языке C++.
80. Типы данных в языке C#: типы значений и ссылочные типы.
81. Встроенные простые типы данных C# (типы значений)
82. Встроенные ссылочные типы C#: object и string.
83. Класс Console и ввод/вывод.
84. Условные операторы C#.
85. Операторы цикла C#.
86. Операторы перехода C#.
87. Массивы C#, создание и инициализация.
88. Пространства имен и оператор using.
89. Объявление класса, данные-члены и функции-члены, модификаторы доступа.
90. Передача параметров в метод.
91. Перегрузка конструкторов.
92. Свойства C#.
93. Структуры.
94. Механизм перегрузки операций.
95. Перегрузка в C# операций отношения.
96. Ограничения на перегрузку операций.
97. Сравнение типов наследования (реализации и интерфейса).
98. Пример использования наследования реализации.
99. Конструкторы производных классов и передача параметров конструктору базового класса.
100. Виртуальные методы.
101. Абстрактные классы и абстрактные методы.

102. Интерфейсы C#.
103. Концепция указателей на функции.
104. Делегаты: назначение, объявление, пример использования.
105. Модель событий C#.
106. Определение коллекции C#.
107. Массивы-списки (класс ArrayList), назначение, основные методы.
108. Стек (класс Stack), назначение, основные методы.
109. Очередь (класс Queue), назначение, основные методы.
110. Концепция файловой системы и концепция баз данных.
111. Определение базы данных.
112. Понятие системы управления базами данных (СУБД).
113. Понятие банка данных.
114. Требования к банку данных
115. Компоненты банка данных.
116. Пользователи банка данных.
117. Администраторы банка данных и их функции.
118. Преимущества и недостатки банка данных.
119. Классификация баз данных, СУБД и банка данных.
120. Общие понятия даталогического проектирования.
121. Подход к даталогическому проектированию.
122. Факторы, влияющие на проектирование базы данных.
123. Особенности даталогических моделей.
124. Ранние подходы к организации базы данных.
125. Особенности СУБД, построенных на основе инвертированных файлов.
126. Иерархические системы.
127. Сетевые системы.
128. Структуры данных.
129. Манипулирование данными.
130. Ограничения целостности.
131. Общие положения реляционного подхода.

132. Базовые понятия реляционных баз данных.
133. Тип данных.
134. Домен.
135. Отношение.
136. Атрибут.
137. Схема отношения.
138. Кортеж.
139. Схема базы данных.
140. Первичный ключ.
141. Внешний ключ.
142. Связи.
143. Типы связей.
144. Фундаментальные свойства отношений.
145. Получение реляционной схемы из ER-модели.
146. История возникновения языка SQL.
147. Средства определения данных.
148. Типы данных SQL.
149. Операторы создания схемы базы данных.
150. Создание и удаление баз данных.
151. Создание, удаление и изменения структуры таблицы.
152. Операторы создания, удаления и изменения индексов.
153. Другие возможности SQL.
154. Основные этапы развития и классификация программного обеспечения ЭВМ. Структура и функции системного, инструментального и прикладного программного обеспечения.
155. Эволюция, назначение и типы операционных систем.
156. Структура ядра операционной системы и его функции. Утилиты, системные обрабатывающие программы и библиотеки.

157. Функциональные компоненты операционной системы автономного компьютера. Организация взаимодействия прикладных программ с операционной системой через функции API.
 158. Средства аппаратной поддержки операционных систем.
 159. Сущность концепции микроядерной архитектуры, ее достоинства и недостатки. Макроядерные операционные системы.
 160. Реализация прикладных программных сред. Стандартизация системных функций и процедур. Стандарты POSIX.
 161. Классификация ресурсов вычислительной системы и возможности их разделения. Понятие вычислительного процесса.
 162. Поясните разницу понятий физического и виртуального адреса, определите соответствующие адресные пространства.
 163. Поясните цепочку преобразований символьных имен программы в физический адрес, используемый процессором.
 164. Распределение памяти статическими и динамическими разделами.
 165. Распределение памяти перемещаемыми разделами.
 166. Страничный способ организации виртуальной памяти, его достоинства и недостатки. Схема преобразования виртуального адреса в физический при страничной организации памяти.
 167. Сегментный способ организации виртуальной памяти, его достоинства и недостатки. Схема преобразования виртуального адреса в физический при сегментной организации памяти.
 168. Поддержка сегментного способа организации виртуальной памяти в микропроцессорах Pentium. Дескриптор сегмента.
 169. Средства поддержки сегментно-страничного способа организации виртуальной памяти в микропроцессорах Pentium.
 170. Поясните необходимость реализации ввода вывода лишь кодом операционной системы.
 171. Сравните адресуемые пространства: оперативной памяти и ввода-вывода.
46. Раскройте понятие - модель внешнего устройства.

172. Поясните термин порт ввода - вывода.
173. Назначение регистра состояния.
174. Каким устройством и в результате чего устанавливается бит готовности регистра состояния.
175. Опишите существующие режимы управления вводом-выводом.
176. Основные задачи, возлагаемые на супервизор ввода-вывода.
177. Поясните, что такое прямой доступ к памяти. Взаимосвязь системных таблиц ввода-вывода.
178. Логическая структура диска.
179. Структура главной загрузочной записи (MBR, Master Boot Record).
180. Таблица разделов диска и формат ее элементов. Флаг активности и системный код раздела. Первичные и расширенные разделы.
181. Поясните понятие «Файловые системы», их функции и назначение.
182. Способы физической организации файлов: непрерывное размещение, связанный список кластеров и индексов, перечисление номеров кластеров.
183. Общие принципы файловой системы типа FAT.
184. Физическая организация файловой системы NTFS. Структура тома с файловой системой NTFS. Назначение главной таблицы файлов MFT.
185. Основные возможности файловой системы NTFS.
186. Структура файлов и каталогов в NTFS.
187. Способы адресации данных в распределенных системах.
188. Протоколы транспортного уровня TCP и UDP.

5. Порядок, форма и язык проведения вступительного испытания

При приеме на обучение по программе магистратуры 09.04.01 «Информатика и вычислительная техника» вступительное испытание проводится по междисциплинарному экзамену, на котором абитуриент должен дать письменные ответы на четыре вопроса согласно правилам приема.

Вступительное испытание проводится на русском языке.

6. Продолжительность вступительного испытания

Продолжительность вступительного испытания составляет 60 минут.

7. Шкала оценивания

Задания оцениваются от 0 до 5 баллов в зависимости от полноты и правильности ответов.

Итоговая оценка определяется на основании среднего арифметического балла, набранного абитуриентом по каждому из четырех вопросов. Неудовлетворительная оценка по одному из вопросов (ниже 3 баллов) автоматически ведет к неудовлетворительной оценке за экзамен в целом.

Критерии оценки:

Оценка	Критерии
Отлично (5 баллов)	Прекрасное знание рассматриваемого вопроса, с совершенно незначительными неточностями.
Хорошо (4 балла)	Хорошее знание рассматриваемого вопроса, но с некоторыми неточностями.
Удовлетворительно (3 балла)	В целом неплохое знание рассматриваемого вопроса, но с заметными ошибками.
Неудовлетворительно (2 балла)	Самое общее представление о рассматриваемом вопросе, отвечающее лишь минимальным требованиям. Серьезные ошибки.
Неудовлетворительно (0-1 балл)	Полное незнание рассматриваемого вопроса. Грубейшие ошибки.

Минимальное количество баллов, подтверждающее успешное прохождение вступительного испытания, определяется локальным актом вуза (Приказ о утверждении перечня вступительных испытаний при ранжировании

списков поступающих; минимального и максимального количества баллов; информации о формах проведения вступительных испытаний, проводимых вузом самостоятельно).

8. Список литературы

Основная литература

1. Устинов, В.В. Основы алгоритмизации и программирование: учебное пособие / В.В. Устинов. – Ч. 2. – Новосибирск: Новосибирский государственный технический университет, 2013.
2. Петров В.Ю. Информатика. Алгоритмизация и программирование. Часть 1. / учебное пособие: Университет ИТМО. – 2016.
3. Лубашева Т.В. Железко Б.А. Основы алгоритмизации и программирования / учебное пособие: Республиканский институт профессионального образования (РИПО) – 2016.
4. Буцык С.В. Крестников А.С. Рузаков А.А. Вычислительные системы, сети и телекоммуникации Челябинский государственный институт культуры, 2016.
5. Мамоиленко С.Н. Ефимов А.В. Сети ЭВМ и телекоммуникаций Сибирский государственный университет телекоммуникаций и информатики, 2018.
6. Гельбух С.С. Архитектура и организация сетей ЭВМ и телекоммуникаций Саратовский государственный технический университет имени Ю.А. Гагарина, ЭБС АСВ 2015
7. Туманов В.Е. Основы проектирования реляционных баз, данных Интернет Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.
8. Баженова И.Ю. Основы проектирования приложений баз данных Интернет Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.

9. Одинокое В.В., Хабибулина Н.Ю. Автоматизированные информационно-управляющие системы Томский государственный университет систем управления и радиоэлектроники, 2014.
10. Филиппов М.В. Стрельников О.И. Вычислительные системы, сети и телекоммуникации Волгоградский институт бизнеса, 2014.
11. Мамоиленко С.Н., Молдованова О.В. Операционные системы. Часть 1. Операционная система Linux Сибирский государственный университет телекоммуникаций и информатики, 2012.

Дополнительная литература

12. Кирнос В.Н. Информатика 2. Основы алгоритмизации и программирования на языке C++. Томский государственный университет систем управления и радиоэлектроники, Эль Контент.
13. Воронцов Ю.А. Ерохин А.Г. Разработка Windows приложений в среде программирования Visual Studio.Net. Московский технический университет связи и информатики. 2016 учебно-методическое пособие.
14. Зиангирова Л.Ф. Вычислительные системы, сети и телекоммуникации Вузовское образование 2015
15. Платунова С.М. Применение межсетевых экранов фирмы ZyXEL в корпоративных сетях Университет ИТМО 2015.
16. Самуйлов С.В. Базы данных Вузовское образование, 2016.
17. Ковган Н.М. Компьютерные сети Республиканский институт профессионального образования (РИПО), 2019.